

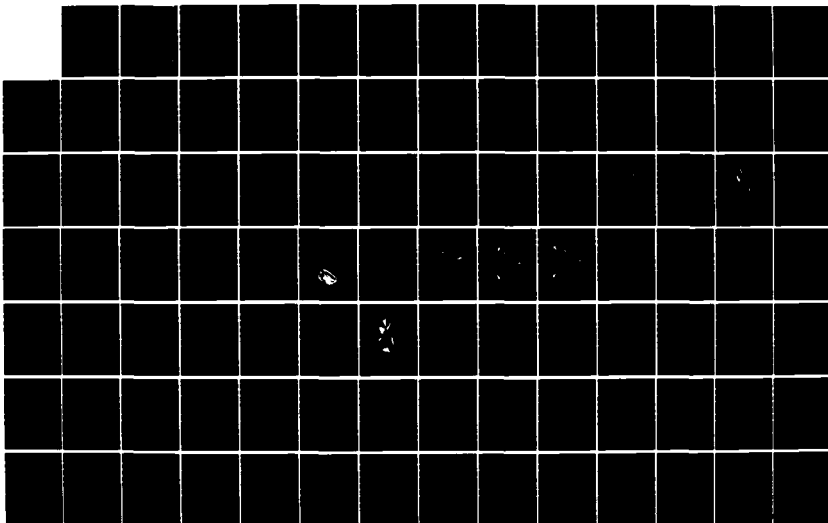
AD-A164 817

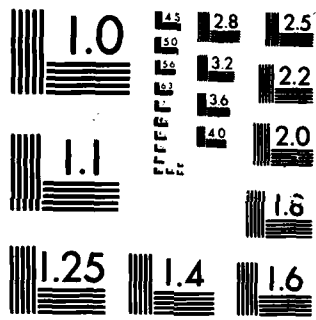
MULTIVARIABLE CONTROL LAW DESIGN FOR ENHANCED AIR
COMBAT MANEUVERING F-15. (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. K A SHEEHAN
DEC 85 AFIT/GE/EE/85D-38 F/G 1/2

1/5

UNCLASSIFIED

NL

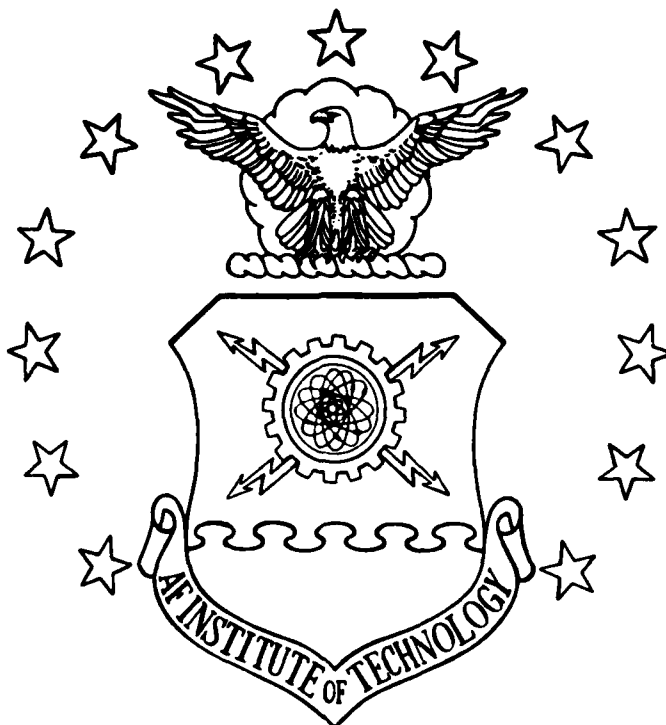




MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A164 017

DTIC FILE COPY



MULTIVARIABLE CONTROL LAW DESIGN
FOR ENHANCED AIR COMBAT MANEUVERING:
F-15/STOL DERIVATIVE FIGHTER

THESIS

Kevin A. Sheehan
Captain, USAF

AFIT/GE/EE/85D-38

DTIC
ELECTE
FEB 13 1986

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

This document has been approved
for public release and sale; its
distribution is unlimited.

86 2 12 034

AFIT/GE/EE/85D-38

MULTIVARIABLE CONTROL LAW DESIGN
FOR ENHANCED AIR COMBAT MANEUVERING:
F-15/STOL DERIVATIVE FIGHTER

THESIS

Kevin A. Sheehan
Captain, USAF

AFIT/GE/EE/85D-38

Approved for public release; distribution unlimited

AFIT/GE/EE/85D-38

MULTIVARIABLE CONTROL LAW DESIGN
FOR ENHANCED AIR COMBAT MANEUVERING:
F-15/STOL DERIVATIVE FIGHTER

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering

Kevin A. Sheehan, B.S.E.E.
Captain, USAF

December 1985

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Date _____	
Approved _____	
Dist _____	
A-1	

Approved for public release; distribution unlimited



Preface

The emphasis of this thesis is in the preliminary design of longitudinal multivariable control laws for the USAF F-15/STOL demonstrator aircraft. The design techniques are based on the work of Professor Brian Porter from the University of Salford, England.

Control law development using the F-15/STOL as a model was particularly satisfying to me given my experience at the controls of the world's greatest fighter, the F-15 Eagle. I would like to thank Captain Greg Mandt of the Flight Dynamics Lab for sponsoring this thesis effort and providing the necessary assistance during the model development phase.

I would also like to express my sincere appreciation to Professors John J. D'Azzo, Chairman of the Electrical and Computer Engineering Department, and Delmar W. Breuer of the Mechanical and Aerospace Engineering Department, for their guidance and tireless scrutiny of this research effort.

This study was conducted in parallel with four fellow Masters students. The ideas that developed from our group discussions were invaluable in achieving the results

of this thesis. My sincere thanks and best wishes to Captains Bruce Acker and Greg Gross, and Lieutenants Bruce Clough and Bob Houston.

Finally, I would like to express my love and appreciation to my wife, Sharon, and my children, Erin and Ryan, for their patience and support throughout this arduous ordeal.

— Kevin A. Sheehan

Table of Contents

	Page
Preface	ii
List of Figures	vii
List of Tables	xvi
List of Symbols	xviii
Abstract	xxiv
I. Introduction	1
1.1 Background	1
1.2 Problem	4
1.3 Approach	5
1.4 Additional Goals	6
1.5 Overview	7
II. The F-15/STOL Aircraft	9
2.1 General Description	9
2.2 Canards	11
2.3 Flaperons	16
2.4 Ailerons	16
2.5 Stabilators	18
2.6 Thrust Vectoring	18
2.7 Summary	25
III. The F-15/STOL Mathematical Model	27
3.1 Introduction	27
3.2 Assumptions	28
Discussion of Assumptions	29
3.3 Model Derivation	31
Equations of Motion	32
State-Space Form	39
Reduced Order Model	42
Actuator Dynamics	47
Sensor Dynamics	52
3.4 Limitations	55
3.5 Model Nonlinearities	57
3.6 Simulation Maneuvers	59
3.7 Summary	61

	Page
IV. Longitudinal Controller Design Method	63
4.1 Introduction	63
4.2 Controllability and Observability	64
4.3 Selecting the Output Vector	65
Transmission Zeros	69
4.4 Design Approach	71
Maximum Maneuver Capability	71
Achieving Stability	77
Tailoring the Input Responses	77
Tailoring the Output Responses	83
Model Development	85
4.5 Parameter Variation	86
4.6 Noise Effects	89
4.7 Summary	90
V. Longitudinal Control Law Design Results	92
5.1 Introduction	92
5.2 Direct Climb (0.3 Mach/FL 200)	93
5.3 Vertical Translation (0.9 Mach/ FL 200)	108
5.4 Pitch Pointing (1.4 Mach/FL 200)	121
5.5 Constant g Pull-Up (2.0 Mach/ FL 400)	135
2.0 g Command Input	137
9.0 g Command Input	154
5.6 Parameter Variation Results	160
Single Controller Analysis	160
Controller Sensitivity	166
5.7 Sensor Noise Results	180
5.8 Summary	192
VI. Conclusions and Recommendations	194
6.1 Design Results	194
6.2 Design Process Improvements	195
6.3 MULTI Improvements	197
6.4 Future Research	201
Index for Appendix A	202
Appendix A: Additions to MULTI	204
Appendix B: Multivariable Control Theory	265
Appendix C: Aero Data and State Space Matrices	278
Appendix D: Design Parameters and Response Plots	298

	Page
Bibliography	369
Vita	371

List of Figures

Figure		Page
2.1	F-15/STOL Aircraft	10
2.2	Control Surface Modifications	12
2.3	Increased Load Factor Capability	14
2.4	Proposed Canard Scheduling	15
2.5	Nozzle Design	19
2.6	Nozzle: Primary Jet Vectoring	21
2.7	Nozzle: Rotating Vane Vectoring	22
2.8	Nozzle: Thrust Reversal	23
2.9	Improved Pitching Moment	24
2.10	Improved Rolling Moment	26
3.1	Aircraft Body Axis System	34
3.2	Stabilator/Canard Actuator Time Response . .	48
3.3	Stabilator/Canard Actuator Frequency Response	48
3.4	Stabilator-Nozzle Actuator Time Response . .	51
3.5	Stabilator-Nozzle Actuator Frequency Response	51
3.6	Digital Block Diagram Configuration	53
3.7	Output Angle Sensor Time Response	54
3.8	Output Angle Sensor Frequency Response . . .	54
4.1a	Pitch Pointing: Linear Simulation (0.9M, FL200)	75
4.1b	Pitch Pointing: Linear Simulation (0.9M, FL200)	75

Figure		Page
4.2a	Pitch Pointing: Nonlinear Simulation (0.9M, FL200)	76
4.2b	Pitch Pointing: Nonlinear Simulation (0.9M, FL200)	76
4.3a	Direct Climb: High Integral Gain, $K_I=6.67 \cdot K_0$ (0.3M, FL200)	81
4.3b	Direct Climb: High Integral Gain, $K_I=6.67 \cdot K_0$ (0.3M, FL200)	81
4.4a	Direct Climb: Low Integral Gain, $K_I=.001 \cdot K_0$ (0.3M, FL200)	82
4.4b	Direct Climb: Low Integral Gain, $K_I=.001 \cdot K_0$ (0.3M, FL200)	82
4.5a	Pitch Pointing: Transmission Zero $= -3.33$ (0.3M, FL200)	84
4.5b	Pitch Pointing: Transmission Zero $= -0.476$ (0.3M, FL200)	84
5.1	Direct Climb: Basic Plant (0.3M/FL200) . . .	98
5.2	Direct Climb: Plant+Actuators (0.3M/FL200). .	98
5.3	Direct Climb: Plant+Actuators+Delay (0.3M/FL200)	99
5.4	Direct Climb: Plant+Actuators+Delay +Sensors (0.3M/FL200)	99
5.5	Direct Climb: Basic Plant (0.3M/FL200) . . .	100
5.6	Direct Climb: Plant+Actuators (0.3M/FL200) .	100
5.7	Direct Climb: Plant+Actuators+Delay (0.3M/FL200)	101
5.8	Direct Climb: Plant+Actuators+Delay +Sensors (0.3M/FL200)	101
5.9	Direct Climb: Basic Plant (0.3M/FL200) . . .	102
5.10	Direct Climb: Plant+Actuators (0.3M/FL200) .	102

Figure		Page
5.11	Direct Climb: Plant+Actuators+Delay (0.3M/FL200)	103
5.12	Direct Climb: Plant+Actuators+Delay +Sensors (0.3M/FL200)	103
5.13	Direct Climb: Basic Plant (0.3M/FL200) . . .	104
5.14	Direct Climb: Plant+Actuators (0.3M/FL200) .	104
5.15	Direct Climb: Plant+Actuators+Delay (0.3M/FL200)	105
5.16	Direct Climb: Plant+Actuators+Delay +Sensors (0.3M/FL200)	105
5.17	Basic Plant (0.3M/FL200)	106
5.18	Direct Climb: Plant+Actuators (0.3M/FL200) .	106
5.19	Direct Climb: Plant+Actuators+Delay (0.3M/FL200)	107
5.20	Direct Climb: Plant+Actuators+Delay +Sensors (0.3M/FL200)	107
5.21	Vertical Trans: Basic Plant (0.9M/FL200) . .	113
5.22	Vertical Trans: Plant+Actuators (0.9M/FL200)	113
5.23	Vertical Trans: Plant+Actuators+Delay (0.9M/FL200)	114
5.24	Vertical Trans: Plant+Actuators+Delay +Sensors (0.9M/FL200)	114
5.25	Vertical Trans: Basic Plant (0.9M/FL200) . .	115
5.26	Vertical Trans: Plant+Actuators (0.9M/FL200)	115
5.27	Vertical Trans: Plant+Actuators+Delay (0.9M/FL200)	116
5.28	Vertical Trans: Plant+Actuators+Delay +Sensors (0.9M/FL200)	116

Figure		Page
5.29	Vertical Trans: Basic Plant (0.9M/FL200) . .	117
5.30	Vertical Trans: Plant+Actuators (0.9M/FL200)	117
5.31	Vertical Trans: Plant+Actuators+Delay (0.9M/FL200)	118
5.32	Vertical Trans: Plant+Actuators+Delay +Sensors (0.9M/FL200)	118
5.33	Vertical Trans: Basic Plant (0.9M/FL200) . .	119
5.34	Vertical Trans: Plant+Actuators (0.9M/FL200)	119
5.35	Vertical Trans: Plant+Actuators+Delay (0.9M/FL200)	120
5.36	Vertical Trans: Plant+Actuators+Delay +Sensors (0.9M/FL200)	120
5.37	Pitch Pointing: Basic Plant (1.4M/FL200) . .	127
5.38	Pitch Pointing: Plant+Actuators (1.4M/FL200)	127
5.39	Pitch Pointing: Plant+Actuators+Delay (1.4M/FL200)	128
5.40	Pitch Pointing: Plant+Actuators+Delay +Sensors (1.4M/FL200)	128
5.41	Pitch Pointing: Basic Plant (1.4M/FL200) . .	129
5.42	Pitch Pointing: Plant+Actuators (1.4M/FL200)	129
5.43	Pitch Pointing: Plant+Actuators+Delay (1.4M/FL200)	130
5.44	Pitch Pointing: Plant+Actuators+Delay +Sensors (1.4M/FL200)	130
5.45	Pitch Pointing: Basic Plant (1.4M/FL200) . .	131
5.46	Pitch Pointing: Plant+Actuators (1.4M/FL200)	131

Figure		Page
5.47	Pitch Pointing: Plant+Actuators+Delay (1.4M/FL200)	132
5.48	Pitch Pointing: Plant+Actuators+Delay +Sensors (1.4M/FL200)	132
5.49	Pitch Pointing: Basic Plant (1.4M/FL200) . . .	133
5.50	Pitch Pointing: Plant+Actuators (1.4M/FL200)	133
5.51	Pitch Pointing: Plant+Actuators+Delay (1.4M/FL200)	134
5.52	Pitch Pointing: Plant+Actuators+Delay +Sensors (1.4M/FL200)	134
5.53	2g Pull-Up: Basic Plant (2.0M/FL400)	142
5.54	2g Pull-Up: Plant+Actuators (2.0M/FL200) . .	142
5.55	2g Pull-Up: Plant+Actuators+Delay (2.0M/FL400)	143
5.56	2g Pull-Up: Plant+Actuators+Delay +Sensors (2.0M/FL400)	143
5.57	2g Pull-Up: Basic Plant (2.0M/FL400)	144
5.58	2g Pull-Up: Plant+Actuators (2.0M/FL400) . .	144
5.59	2g Pull-Up: Plant+Actuators+Delay (2.0M/FL400)	145
5.60	2g Pull-Up: Plant+Actuators+Delay +Sensors (2.0M/FL400)	145
5.61	2g Pull-Up: Basic Plant (2.0M/FL400)	146
5.62	2g Pull-Up: Plant+Actuators (2.0M/FL400) . .	146
5.63	2g Pull-Up: Plant+Actuators+Delay (2.0M/FL400)	147
5.64	2g Pull-Up: Plant+Actuators+Delay +Sensors (2.0M/FL400)	147
5.65	2g Pull-Up: Basic Plant (2.0M/FL400)	148

Figure		Page
5.66	2g Pull-Up: Plant+Actuators (2.0M/FL400) . .	148
5.67	2g Pull-Up: Plant+Actuators+Delay (2.0M/FL400)	149
5.68	2g Pull-Up: Plant+Actuators+Delay +Sensors (2.0M/FL400)	149
5.69	2g Pull-Up: Basic Plant (2.0M/FL400)	150
5.70	2g Pull-Up: Plant+Actuators (2.0M/FL400) . .	150
5.71	2g Pull-Up: Plant+Actuators+Delay (2.0M/FL400)	151
5.72	2g Pull-Up: Plant+Actuators+Delay +Sensors (2.0M/FL400)	151
5.73	2g Pull-Up: Basic Plant (2.0M/FL400)	152
5.74	2g Pull-Up: Plant+Actuators (2.0M/FL400) . .	152
5.75	2g Pull-Up: Plant+Actuators+Delay (2.0M/FL400)	153
5.76	2g Pull-Up: Plant+Actuators+Delay +Sensors (2.0M/FL400)	153
5.77	9g Pull-Up: Basic Plant (2.0M/FL400)	157
5.78	9g Pull-Up: Basic Plant (2.0M/FL400)	157
5.79	9g Pull-Up: Basic Plant (2.0M/FL400)	158
5.80	9g Pull-Up: Basic Plant (2.0M/FL400)	158
5.81	9g Pull-Up: Basic Plant (2.0M/FL400)	159
5.82	9g Pull-Up: Basic Plant (2.0M/FL400)	159
5.83	Pitch Pointing: Single Controller Analysis (1.4M/FL200)	163
5.84	Vertical Trans: Single Controller Analysis (1.4M/FL200)	163
5.85	Direct Climb: Single Controller Analysis (1.4M/FL200)	164

Figure		Page
5.86	Direct Climb: Single Controller Analysis (2.0M/FL400)	164
5.87	Pitch Pointing: Single Controller Analysis (2.0M/FL400)	165
5.88	Vertical Trans: Single Controller Analysis (2.0M/FL400)	165
5.89	Vertical Trans: +9% M-Delta(C) , (0.9M/FL200)	168
5.90	Vertical Trans: +9% M-Delta(C) , (0.9M/FL200)	168
5.91	Vertical Trans: +9% M-Delta(C) , (0.9M/FL200)	169
5.92	Vertical Trans: +9% M-Delta(C) , (0.9M/FL200)	169
5.93	Vertical Trans: -5% M-Delta(C) , (0.9M/FL200)	170
5.94	Vertical Trans: -5% M-Delta(C) , (0.9M/FL200)	170
5.95	Vertical Trans: -5% M-Delta(C) , (0.9M/FL200)	171
5.96	Vertical Trans: -5% M-Delta(C) , (0.9M/FL200)	171
5.97	Vertical Trans: +7% M-Delta(H) , (0.9M/FL200)	172
5.98	Vertical Trans: +7% M-Delta(H) , (0.9M/FL200)	172
5.99	Vertical Trans: +7% M-Delta(H) , (0.9M/FL200)	173
5.100	Vertical Trans: +7% M-Delta(H) , (0.9M/FL200)	173
5.101	Vertical Trans: -25% M-Delta(H) , (0.9M/FL200)	174

Figure		Page
5.102	Vertical Trans: -25% M-Delta(H), (0.9M/FL200)	174
5.103	Vertical Trans: -25% M-Delta(H), (0.9M/FL200)	175
5.104	Vertical Trans: -25% M-Delta(H), (0.9M/FL200)	175
5.105	Vertical Trans: +100% X-Delta(H), (0.9M/FL200)	176
5.106	Vertical Trans: +100% X-Delta(H), (0.9M/FL200)	176
5.107	Vertical Trans: +100% X-Delta(H), (0.9M/FL200)	177
5.108	Vertical Trans: +100% X-Delta(H), (0.9M/FL200)	177
5.109	Vertical Trans: -50% X-Delta(H), (0.9M/FL200)	178
5.110	Vertical Trans: -50% X-Delta(H), (0.9M/FL200)	178
5.111	Vertical Trans: -50% X-Delta(H), (0.9M/FL200)	179
5.112	Vertical Trans: -50% X-Delta(H), (0.9M/FL200)	179
5.113	Pitch Pointing: No Sensor Noise (1.4M/FL200)	184
5.114	Pitch Pointing: No Sensor Noise (1.4M/FL200)	184
5.115	Pitch Pointing: Typical Sensor Noise (1.4M/FL200)	185
5.116	Pitch Pointing: Typical Sensor Noise (1.4M/FL200)	185
5.117	Pitch Pointing: 2.5X Typical Sensor Noise (1.4M/FL200)	186

Figure		Page
5.118	Pitch Pointing: 2.5X Typical Sensor Noise (1.4M/FL200)	186
5.119	Pitch Pointing: 5X Typical Sensor Noise (1.4M/FL200)	187
5.120	Pitch Pointing: 5X Typical Sensor Noise (1.4M/FL200)	187
5.121	Pitch Pointing: No Sensor Noise (1.4M/FL200)	188
5.122	Pitch Pointing: No Sensor Noise (1.4M/FL200)	188
5.123	Pitch Pointing: Typical Sensor Noise (1.4M/FL200)	189
5.124	Pitch Pointing: Typical Sensor Noise (1.4M/FL200)	189
5.125	Pitch Pointing: 2.5X Typical Sensor Noise (1.4M/FL200)	190
5.126	Pitch Pointing: 2.5X Typical Sensor Noise (1.4M/FL200)	190
5.127	Pitch Pointing: 5X Typical Sensor Noise (1.4M/FL200)	191
5.128	Pitch Pointing: 5X Typical Sensor Noise (1.4M/FL200)	191
A.1	Sample Custom Input--Unsmoothed	225
A.2	Sample Custom Input--Smoothed	225
A.3	Option #22 Algorithm Outline	226
B.1	System Block Diagram--Continuous Case	268
B.2	System Block Diagram--Discrete Case	270

List of Tables

Table	Page
3.1 Open-Loop Plant Eigenvalues	42
3.2 Actuator Model FOM Comparison	49
3.3 Sensor Model FOM Comparison	55
3.4 Control Input Deflection/Rate Limits	56
3.5 Simulation Maneuvers and Flight Conditions .	60
4.1 Maximum Maneuver Capability	73
5.1 Design Parameters and Controller Matrices . .	96
5.2 Design Output Figures of Merit	97
5.3 Design Parameters and Controller Matrices . .	111
5.4 Design Output Figures of Merit	112
5.5 Design Parameters and Controller Matrices . .	125
5.6 Design Output Figures of Merit	126
5.7 Design Parameters and Controller Matrices . .	140
5.8 Design Output Figures of Merit	141
5.9 Design Parameters and Controller Matrices . .	155
5.10 Design Output Figures of Merit	156
B.1 Asymptotic Equations for Zero- B_2 Form	273
C.1 Longitudinal State Space Matrices (Four-State Model)	279
C.2 Longitudinal State Space Matrices (Four-State Model)	280
C.3 Longitudinal State Space Matrices (Three-State Model)	281

Table		Page
C.4	Longitudinal State Space Matrices (Four-State Model)	282
C.5	Longitudinal State Space Matrices (Three-State Model)	283
C.6	Longitudinal State Space Matrices (Four-State Model)	284
C.7	Longitudinal State Space Matrices (Three-State Model)	285
D.1	Design Parameters and Controller Matrices . .	299
D.2	Design Parameters and Controller Matrices . .	305
D.3	Design Parameters and Controller Matrices . .	311
D.4	Design Parameters and Controller Matrices . .	317
D.5	Design Parameters and Controller Matrices . .	322
D.6	Design Parameters and Controller Matrices . .	327
D.7	Design Parameters and Controller Matrices . .	332
D.8	Design Parameters and Controller Matrices . .	337
D.9	Design Parameters and Controller Matrices . .	342
D.10	Design Parameters and Controller Matrices . .	347
D.11	Design Parameters and Controller Matrices . .	354
D.12	Design Parameters and Controller Matrices . .	358
D.13	Design Parameters and Controller Matrices . .	365

List of Symbols

\underline{A}	Continuous-time plant matrix
α_{ts}	Trim angle of attack; stability axis
$A_{n_{cg}}$	Longitudinal acceleration at the center of gravity
A_{n_p}	Longitudinal acceleration at the cockpit
AOA	Angle of attack
α	Angle of attack, perturbation angle of attack in perturbation equations
α_T	Trim angle of attack
$\bar{\alpha}$	Ratio of proportional to integral feedback
\underline{B}	Continuous-time plant matrix
b	Wing span
BW	Bandwidth
\underline{C}	Continuous-time Output Matrix
c	Mean Aerodynamic Cord
CG, cg	Center of gravity
C_m	Nondimensional coefficient of pitching moment
C_{m_α}	Nondimensional variation of pitching moment with angle of attack
C_{m_δ}	Nondimensional variation of pitching moment with stabilator (δ_H), canard (δ_C) or stab-nozzle (δ_{HN})
C_{m_q}	Nondimensional variation of pitching moment with pitch rate
C_{m_u}	Nondimensional variation of pitching moment with forward velocity perturbations
cos	Cosine

C_x	Nondimensional x-force coefficient
C_{x_α}	Nondimensional variation of x-force with angle of attack
C_{x_δ}	Nondimensional variation of x-force with stabilator (δ_H), canard (δ_C), throttle (δ_T), or stab-nozzle (δ_{HN})
C_{x_q}	Nondimensional variation of x-force with pitch rate
C_{x_u}	Nondimensional variation of x-force with forward velocity perturbation
C_z	Nondimensional z-force coefficient
C_{z_α}	Nondimensional variation of z-force with angle of attack
$C_{z_{\dot{\alpha}}}$	Nondimensional variation of z-force with the rate of change of angle of attack
C_{z_δ}	Nondimensional variation of z-force with stabilator (δ_H), canard (δ_C), or stab-nozzle (δ_{HN})
C_{z_q}	Nondimensional variation of z-force with pitch rate
C_{z_u}	Nondimensional variation of z-force with forward velocity perturbation
deg	Degree
δ_C	Canard deflection
δ_H	Stabilator deflection
δ_{HN}	Nozzle deflection
δ_N	Nozzle deflection
$\underline{e(t)}, \underline{e}$	Error vector
$\underline{e(kT)}$	Discrete error vector
ϵ	Epsilon scalar multiplier
FL	Flight level (constant pressure altitude)

ft	Feet
F_z	Force in z-body axis
$F_{z_{cg}}$	z-force at center of gravity
f	Sampling frequency
g	Gravity, type of pilot command, gain constant
$\underline{G}(s)$	Transfer function matrix
I_{xx}	Moment of inertia about x-axis
I_{yy}	Moment of inertia about y-axis
I_{zz}	Moment of inertia about z-axis
I_{xz}	Product of inertia about xz-axes
\underline{I}	Identity matrix
\underline{K}_0	Proportional control law feedback matrix
\underline{K}_1	Integral control law feedback matrix
Lat-Dir	Lateral-Directional
lbs	pounds
Long	Longitudinal
ℓ	Number of system outputs
ℓ_x	Distance from CG to sensor location along x-axis
\underline{M}	Measurement matrix
m	Aircraft mass, number of inputs
M_α	Dimensional variation of pitching moment with angle of attack
$M_{\dot{\alpha}}$	Dimensional variation of pitching moment with the rate of change of angle of attack
M_c	Controllability matrix
M_o	Observability matrix

M_q	Dimensional variation of pitching moment with pitch rate
M_θ	Dimensional variation of pitching moment with pitch angle
M_x	Moment about the x-axis
n	Number of states
p	Number of outputs, roll rate
q	Pitch rate
\bar{q}	Dynamic pressure
r	Yaw rate
rad	Radians
S	Surface area
s	Laplace operator
sec	Seconds
sin	Sine
σ	Elements of the Sigma (Σ) matrix
Σ	Sigma Gain Weighting matrix
\underline{T}	Transformation matrix
T	Sampling period, Thrust
U	Velocity along x-axis
u	Perturbation velocity along x-axis
\underline{u}	Input vector
V	Velocity along y-axis
v	Perturbation velocity along y-axis
\underline{v}	Command input vector
V_T	Forward Velocity

W	Velocity along z-axis, aircraft weight
w	Perturbation velocity along z-axis
\underline{w}	Controller output vector
\underline{x}	State vector
X_α	Dimensional variation of x-force with angle of attack
$X_{\dot{\alpha}}$	Dimensional variation of x-force with the rate of change of angle of attack
X_δ	Dimensional variation of x-force with stabilator (δ_H), canard (δ_C), throttle (δ_T), or stab-nozzle (δ_{HN})
X_q	Dimensional variation of x-force with pitch rate
X_u	Dimensional variation of x-force with forward velocity perturbation
\underline{y}	Output vector
\underline{y}_{fb}	Feedback vector (after sensor measurement)
\underline{y}_{out}	Feedback vector (before sensor measurement)
$\underline{z}(t), \underline{z}$	Integral of error vector
$\underline{z}(kT)$	Discrete Integral of error vector
Z_α	Dimensional variation of z-force with angle of attack
$Z_{\dot{\alpha}}$	Dimensional variation of z-force with the rate of change of angle of attack
Z_δ	Dimensional variation of z-force with stabilator (δ_H), canard (δ_C), or stab-nozzle (δ_{HN})
Z_q	Dimensional variation of z-force with pitch rate
Z_u	Dimensional variation of z-force with forward velocity perturbation
Z_t	Transmission zeros
$Z_{1,2}$	Finite system roots

z_3	Infinite system roots
θ	Pitch angle
θ_c	Pitch angle command
θ_e	Pitch angle error
ϕ	Roll angle
ψ	Yaw angle
$\Gamma(\lambda)$	Asymptotic transfer function matrix
$\hat{\Gamma}(\lambda)$	Asymptotic transfer function matrix (fast roots)
$\tilde{\Gamma}(\lambda)$	Asymptotic transfer function matrix (slow roots)

Abstract

Digital flight control laws which demonstrate improved air combat maneuverability are developed for the F-15/STOL (Short Take-Off and Landing) derivative fighter. Proportional plus integral controllers are designed for the longitudinal mode using a multivariable control law theory developed by Professor Brian Porter of the University of Salford, England. Control laws are formulated by use of a computer-aided, multivariable design program entitled MULTI. In addition, MULTI performs a digital closed-loop simulation for controller performance analysis.

The aircraft model is developed from linearized data provided by McDonnell Aircraft Co., the prime contractor for the F-15/STOL. Canard and thrust vectoring technology, in addition to conventional control surfaces, are included in the model. Decoupling of the longitudinal output variables is achieved and demonstrated by four maneuvers (pitch-pointing, vertical translation, direct climb, and constant g pull-up). Plant parameter variation effects are also examined. Destabilizing effects to include actuator and sensor dynamics, computational time delay, random Gaussian sensor noise, and simulation nonlinearities are included.

Results show stable responses for all simulations. Except for the most demanding simulations (all destabilizing effects considered), controller responses are smooth and well behaved.

Recommendations include proposed future work in thrust vector modeling and suggested improvements to the computer-aided design program, MULTI.

MULTIVARIABLE CONTROL LAW DESIGN FOR
ENHANCED AIR COMBAT MANEUVERING:
F-15/STOL DERIVATIVE FIGHTER

I. Introduction

1.1 Background

Air-to-air combat, by definition, is the engagement of two or more aircraft in an aerial "dogfight" with air superiority as the eventual goal. In essence, it is control of the skies above the battlefield. Military strategists have found that control of this airspace is crucial in determining the outcome of the land battle. Without air superiority, the opponent's airpower can defeat even a numerically superior ground force.

Consequently, tomorrow's fighter aircraft must be designed with the capabilities to accurately intercept and quickly destroy enemy aircraft, even when outnumbered by the opponent. Quick kills are important from the survivability standpoint. Prolonged engagements increase the risk of attack by an unseen enemy. Therefore, in order to achieve these goals, aircraft maneuverability must be improved. Given equal pilot skills and weapons, the most maneuverable aircraft has the decided advantage in any air combat scenario.

The desire for enhanced maneuverability has led aircraft designers to reduce the static stability margin of modern fighters such as the F-16. This new concept of relaxed static stability quickens the aircraft's response to control surface inputs. Further improvements in maneuverability have been demonstrated by the AFTI F-16 through the use of redundant control surfaces (2). Ventral canards, positioned forward of the aircraft CG and combined with conventional rudders, can produce direct side forces. The use of direct side forces allows the aircraft to perform turns with virtually zero bank angle or sideslip.

Other aircraft, such as the Grumman X-29, use horizontal canards which are mounted forward of the aircraft CG. These additional surfaces combine with wing trailing edge flaperons to produce a direct force in the plane of symmetry which can decouple flight path angle from aircraft pitch attitude (6). Direct force technology can be more effective than conventional aircraft control for producing a specific aircraft response. For example, when commanding a climb, conventional aircraft use the negative lift produced by the elevator to generate a positive moment. The moment then increases the AOA of the wings to increase lift. Total lift is reduced, however, by the amount used in producing the necessary moment.

Current research in the area of direct force technology has validated its potential benefits for air combat

maneuvering (5). Findings indicate that the limited g capability of direct force, single-plane maneuvering results in limited improvement to defensive evasion during aerial gun tracking maneuvers. It has been shown, however, that the coupling of several single-plane maneuvers (simultaneous direct force in the y and z axes) dramatically increases defensive capability over conventional aircraft (5:28). Offensive potential is also promising.

Aircraft such as the F-15/STOL use a blending of symmetrical canards, ailerons, and stabilators to produce either rotation or translation with a minimal reduction in the total net lift. As a result, maneuverability is increased both in response time and maneuver capability.

An additional feature of the F-15/STOL aircraft is the nozzles that are used to vector the engine thrust either symmetrically or asymmetrically to produce direct forces or moments (16). In addition, upper and lower engine exhaust vanes give an added degree of velocity control to help sustain combat energy levels and increase maneuverability.

Classical control theory falls short when applied to these advanced aircraft designs. Single input-single output (SISO) design procedures do not fully exploit the flexibility and capability of multiple input-multiple output (MIMO) flight control systems. Furthermore, because of the inherent static instability built into aircraft

designs such as the F-15/STOL, digital control systems with fast sampling rates are required to maintain control throughout the flight envelope. In certain critical flight conditions, the pilot cannot respond fast enough or with the proper control inputs which are needed to maintain aircraft control.

Modern control theory has produced several design techniques to solve the MIMO problem. Quantitative Feedback Theory (QFT) employs the advantages of frequency domain analysis and allows for "up front" design of parameter variation. Kalman filter techniques using the Linear Quadratic Generator (LQG) have also proven effective. The design technique used in this thesis was developed by Professor Brian Porter of the University of Salford, England (18). His procedure uses time domain techniques in developing a high gain, error-actuated, proportional plus integral controller. The method is directly applicable to the design of digital controllers. In the digital case, the high gain condition is equivalent to fast sampling rates (18). A complete description of the Porter design method is included in Appendix B.

1.2 Problem

The F-15/STOL exhibits open-loop static instability at certain points in the flight envelope (Chapter III). Because of the instability, the design of an acceptable

control law is a two-fold problem. First, and most importantly, the design must guarantee stability at all flight conditions. Secondly enhanced maneuverability that results from the application of direct force control must be demonstrated in all of the control law designs. This improvement is seen in the decoupling of the output variables.

With these two major goals in mind, this thesis undertakes the design of digitally implemented flight control laws for the F-15/STOL demonstrator aircraft. In addition to stability and improved maneuverability, the design goals include:

1. Model realism to include actuator/sensor dynamics and computational time delay.
2. Adherence to maximum control input deflection limits and rates.
3. Low sensitivity to parameter variation and sensor noise.
4. Single controller capability for all maneuvers at each flight condition.

1.3 Approach

The Porter method is used to design multivariable longitudinal control laws at four selected points in the aircraft's flight envelope (Chapter III). Aeronautical data is provided by McDonnell Aircraft Corp. for use in

this study (15). Both the longitudinal and lateral data are included in Appendix C.

The materials and equipment used for this study consist of the CDC Cyber computer located at Aeronautical Systems Division, Wright-Patterson AFB, Ohio and the supporting software programs of ZERO, TOTAL, and MULTI. The program MULTI allows for the design and simulation of multivariable control laws consistent with the Porter method (13). The simulation maneuvers are chosen to demonstrate the output decoupling effects of the high gain, error-actuated digital controller.

The design process consists of a "building block" approach where the initial design for the basic plant is successively modified as each stage of dynamics is added to the model. A sample design is then analyzed for parameter variation insensitivity and sensor noise effects.

1.4 Additional Goals

Improvements to the MULTI program are also a part of this research effort. The modifications are made as a joint effort with a parallel thesis by Acker (1) which investigates the landing characteristics of the F-15/STOL. The MULTI enhancements are listed as follows:

1. Include disturbance and sensor noise capability within the design simulation.

2. Provide for a Monte Carlo analysis of the noise corrupted results.

3. Introduce the capability to plot a system output as a linear combination of the system states and control inputs.

These modifications are necessary in order to meet the design goals of this study. A detailed explanation of these improvements is done by Acker and included in Appendix A.

1.5 Overview

Chapter II presents an overall description of the F-15/STOL aircraft. Detailed figures show the control improvements of canards and thrust vectoring that are added to the standard F-15 airframe.

In Chapter III, the derivation of the mathematical model is given along with a complete development of the assumptions and limitations that govern its validity. These limitations form a baseline for analysis in later chapters.

The design methodology is presented in Chapter IV. Numerous techniques are offered that are effective in this design. Simulation results are presented where applicable in order to validate the techniques that are given. No attempt is made to establish universal application of the design method. As an alternative, the techniques presented

provide insight for future designers facing similar problems with their own particular model.

The results of the longitudinal control law designs are given in Chapter V. A thorough analysis is presented that includes the effects of parameter variation and sensor noise corruption on the system responses.

The conclusions drawn from the results of the previous two chapters are compiled in Chapter VI. A comparison is presented between the theoretical results obtained from the literature and the results of this study. Finally, the recommendations for future research and improvements to MULTI are included in Chapter VI.

II. The F-15/STOL Aircraft

2.1 General Description

The proposed F-15/STOL derivative fighter is a modified version of the standard F-15/B (two-seat model). Its overall dimensions and control surface modifications are shown in Figure 2.1. From a flight controls standpoint, the two most important modifications are the addition of the canards and the engine mounted thrust vectoring apparatus (Figure 2.2).

The purpose behind the development of this aircraft prototype is twofold. First, to research, develop, and validate the appropriate technologies that provide the STOL capabilities for fighter aircraft. Two of the technologies that this thesis deals with are direct force and thrust vectoring. Secondly, through the validation of these new design techniques, greater design flexibility will be available to future advanced fighters. This aircraft prototype is being designed and built as a technology demonstrator to serve as a forerunner to future, more advanced aircraft designs.

The major performance requirements for the F-15/STOL are:

1. Takeoff/landing capability of 1500 feet on a 50' wide runway surface at night and in adverse weather.

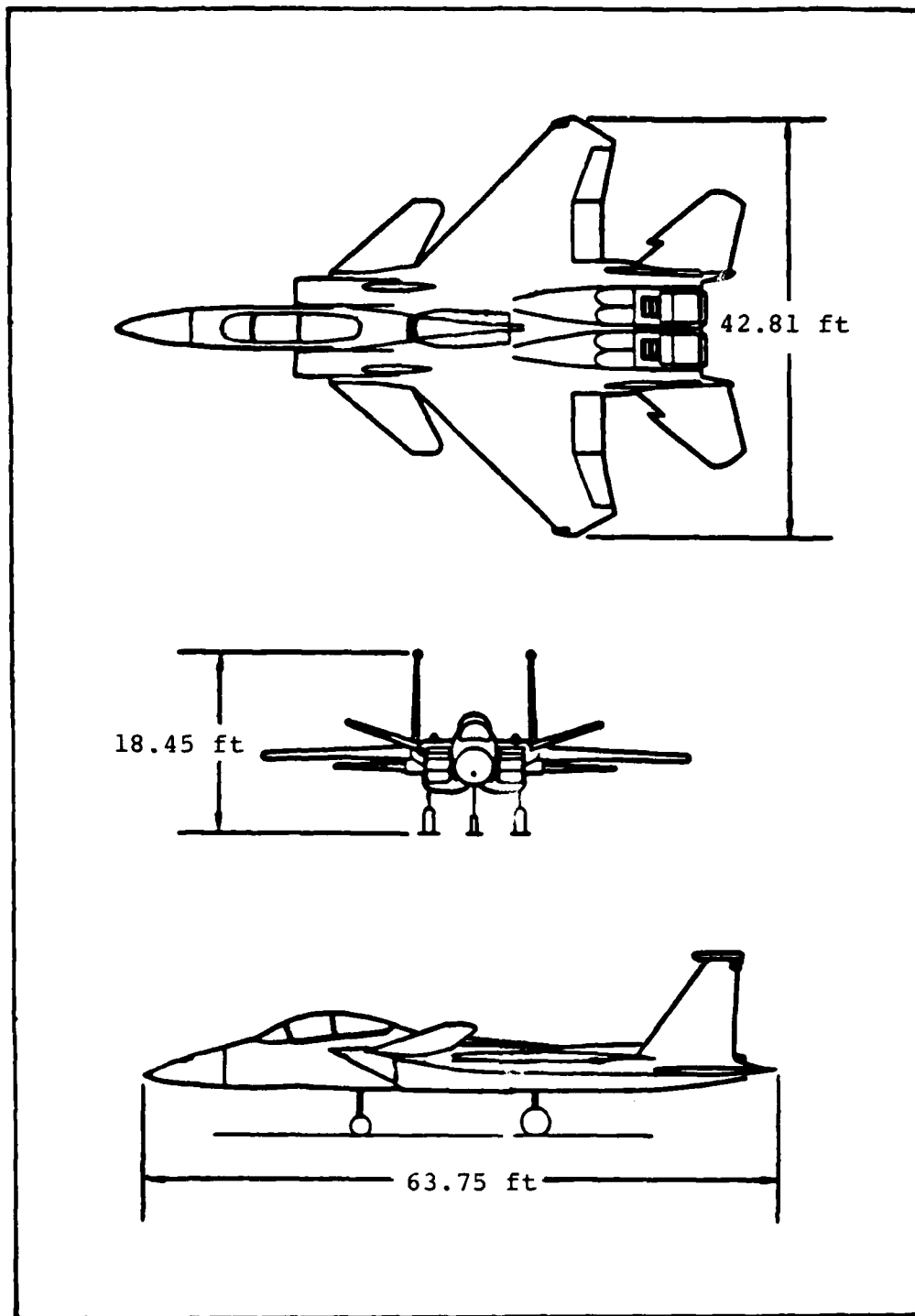


Fig. 2.1. F-15/STOL Aircraft (15)

2. Takeoff in combat configuration with full internal fuel and 6000 lb. external payload.
3. Precision landing with normal fuel reserves.
4. Improved air combat maneuverability over the standard F-15/B.
5. Equal or greater range capability over a similarly configured F-15/B.

This thesis effort is concerned with the design and evaluation of several digitally implemented control laws that demonstrate improved air combat maneuverability while the precision STOL capabilities of this model are investigated in other theses (1). Improved maneuverability is accomplished through the combined use of the following conventional and nonconventional controls.

2.2 Canards

Conventional F-18 horizontal stabilators are used as nonconventional, dihedral canards on the F-15/STOL. The canards are mounted just aft of the cockpit area and outboard of the engine inlets (Figure 2.2). These surfaces can be operated either symmetrically for pitch control or differentially for roll control. The 20 degree dihedral angle gives the additional capability of direct side force when operated differentially. By combining this force forward of the CG with the force produced by the conventional rudders aft of the CG, unconventional maneuvers such as

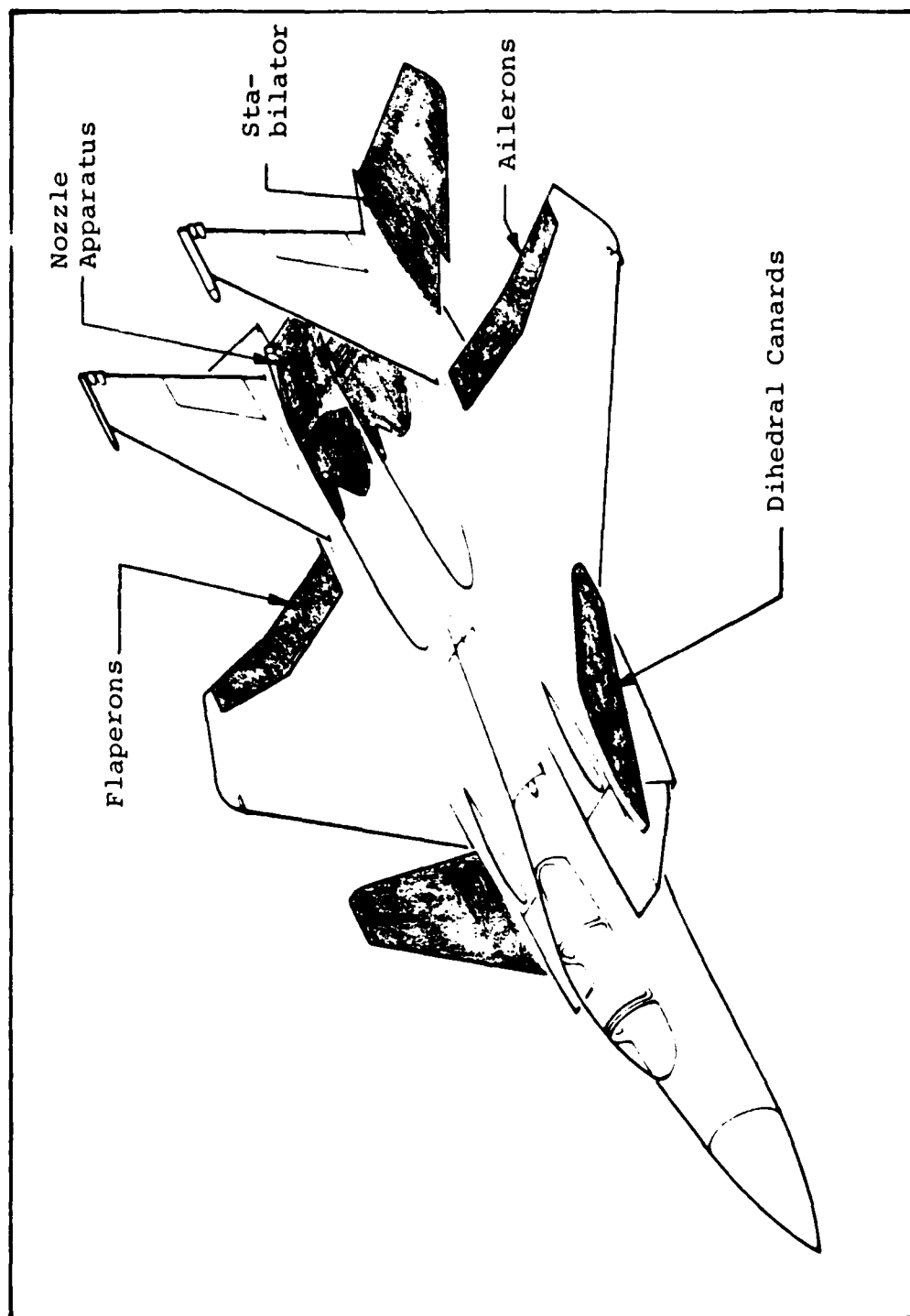


Fig. 2.2. Control Surface Modifications (15)

flat turns with zero bank and sideslip angle can be performed.

The additional benefit of an increased maximum load factor ("g" capability) is realized by aircraft incorporating canard technology (Figure 2.3). The negative lift effect of the tail is offset and reduced by the positive lift of the canard and additional force of the thrust vector. The net effect is an aircraft capable of 9.0 G load factors with essentially the same airframe as the standard F-15.

McDonnell Aircraft Co. (MCAIR), the prime contractor for the F-15/STOL, plans to use canard scheduling as a function of AOA and Mach number (Figure 2.4). As a result, their control law design for the longitudinal mode does not include symmetric canards as an independent control input. This thesis assumes independent control of the canard in an effort to demonstrate the aircraft's maneuver potential. The assumption of independently controlled canards, however, creates an implementation problem of pilot control. This thesis attempts to demonstrate the maneuver potential of the F-15/STOL whereas cockpit implementation is beyond the scope of this study. It is assumed that the implementation problem would be solved if this technology was adopted.

The canard has maximum deflection limits of +15 and -35 degs. The surface is limited to a maximum

Canards Improve Performance

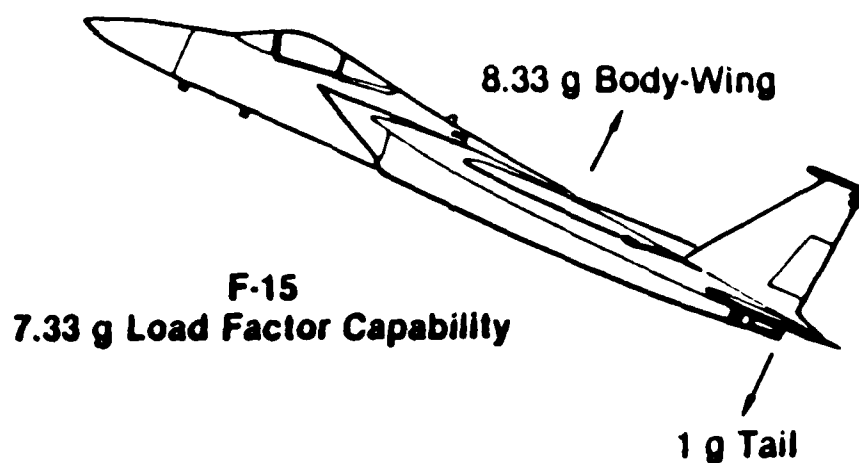
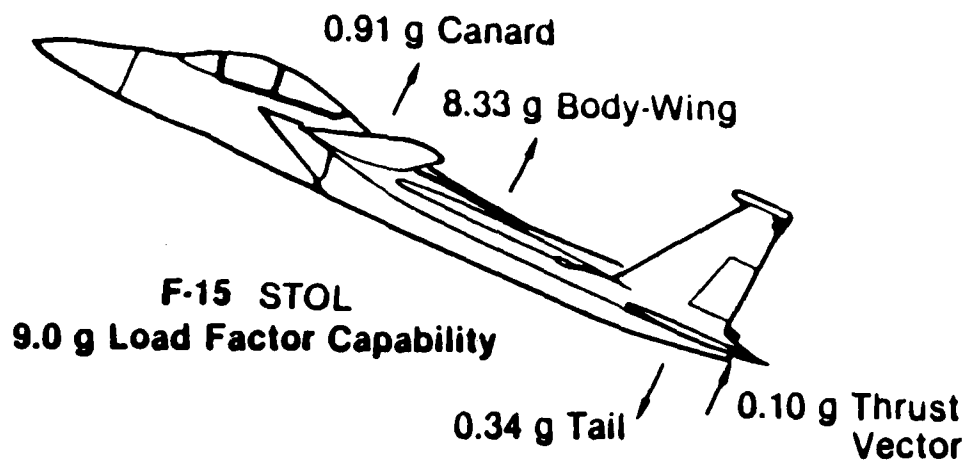


Fig. 2.3. Increased Load Factor Capability (15)

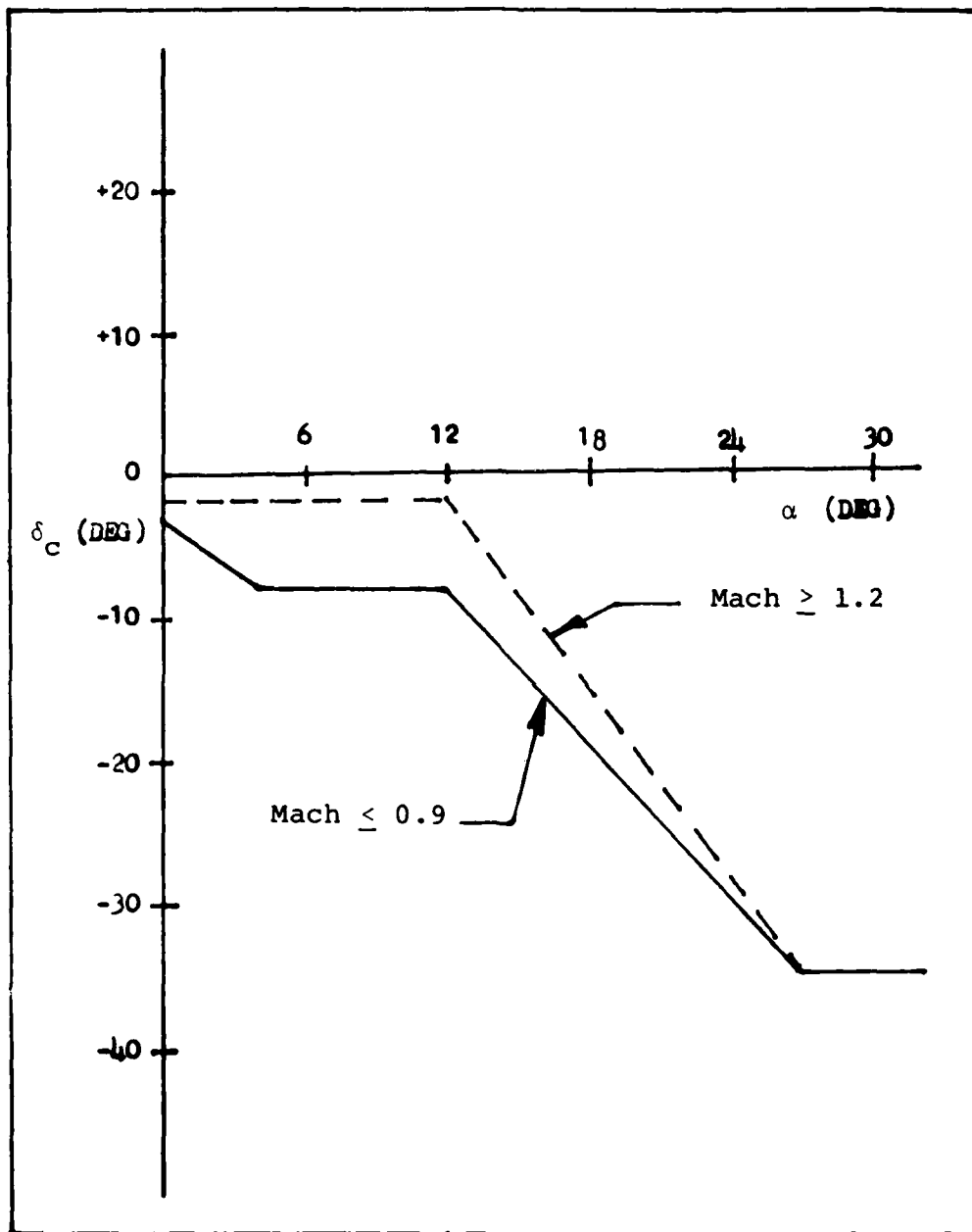


Fig. 2.4. Proposed Canard Scheduling (15)

deflection rate of 23 degs/sec, no-load. Since data for deflection limits and rates as a function of dynamic pressure were not available, the no-load limits are used throughout the flight envelope for this design.

2.3 Flaperons

The conventional flaps of the F-15/B which are manually controlled by the pilot for use during landing approaches have been modified into a dynamic control surface for the F-15/STOL. As with the canards, the flaperons can be moved symmetrically for pitch control or differentially for roll control.

The flaperons are used primarily during landing approach for roll and pitch control while the "drooped" ailerons assume the duty of conventional flaps, deflected symmetrically down to a fixed angle.

The deflection limits (+30, -0 degs) have not been changed from the F-15/B. The zero degree minimum limit prohibits any effective use of this control surface for combat maneuvering. Consequently, the flaperons are set to zero degrees and are not used as control inputs in this study. The no-load rate limit for the flaperons is 100 degs/sec.

2.4 Ailerons

Conventional ailerons can only be deflected differentially for use in the lateral-directional mode to control bank angle and roll rate. At high angles of

attack, the ailerons become less effective and more destabilizing due to airflow separation and drag effects. Modern aircraft such as the F-15, incorporate flight controllers that "washout" pilot inputs to the ailerons at high AOA and rely instead on differential stabilators to accomplish the desired rolling maneuver. At lower AOA, however, their relatively long moment arm from the aircraft's longitudinal axis makes the ailerons a very effective roll control device.

The F-15/STOL takes the flight control concepts of the F-15/B one step further by allowing symmetrical aileron deflection to enhance longitudinal pitch control. In contrast to their rolling authority, however, the ailerons have a relatively short moment arm as measured from the aircraft's lateral axis. As an example, at 0.3 Mach and FL 200, the canard and stabilator are approximately 10 to 20 times more effective in pitch than the symmetrical ailerons (Appendix C). Because of the relative ineffectiveness of the ailerons, they are not used as control inputs in the longitudinal mode for this study. In the lateral mode, the ailerons could be combined with differential stabilator and nozzle to form a very effective, single input for use in commanding bank angles and roll rates.

The ailerons have a maximum deflection limit of ± 20 degs with a rate limit of 100 degs/sec.

2.5 Stabilators

As mentioned earlier, the F-15 stabilators can deflect either differentially for roll control or symmetrically to control longitudinal angles and rates. Their relative size compensates for their short moment arm and makes them a very effective lateral control device. In addition, the stabilator deflection limits of +15 to -29 degs provides a substantial force for use in direct lift maneuvers when combined with the canards and thrust vectoring nozzles.

This thesis uses the stabilators as a primary input in the longitudinal mode. The stabilators have a maximum rate limit of 46 degs/sec, no-load. Again, lacking more exact data, this rate is used throughout the flight envelope and is well suited for the purpose of this study.

2.6 Thrust Vectoring

One of the most important technologies to be explored with the F-15/STOL is the use of thrust vectoring in air-to-air combat. The thrust vectoring apparatus is made up primarily of two separate parts, a two-dimensional nozzle and a rotating vane assembly (Figure 2.5).

The two-dimensional nature of the nozzle results from its restricted up and down motion within the x-z plane of the aircraft. During jet vectoring, the correct exhaust throat area is controlled by the divergent and

Nozzle Design

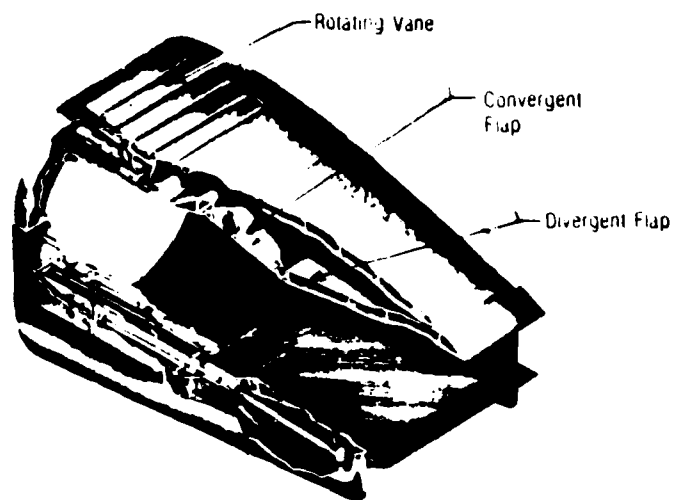
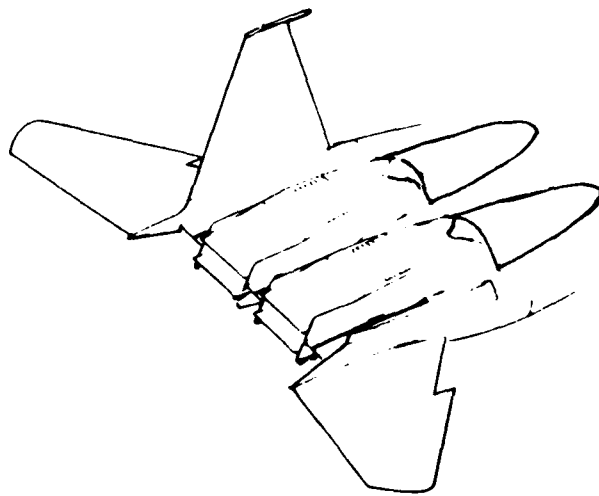


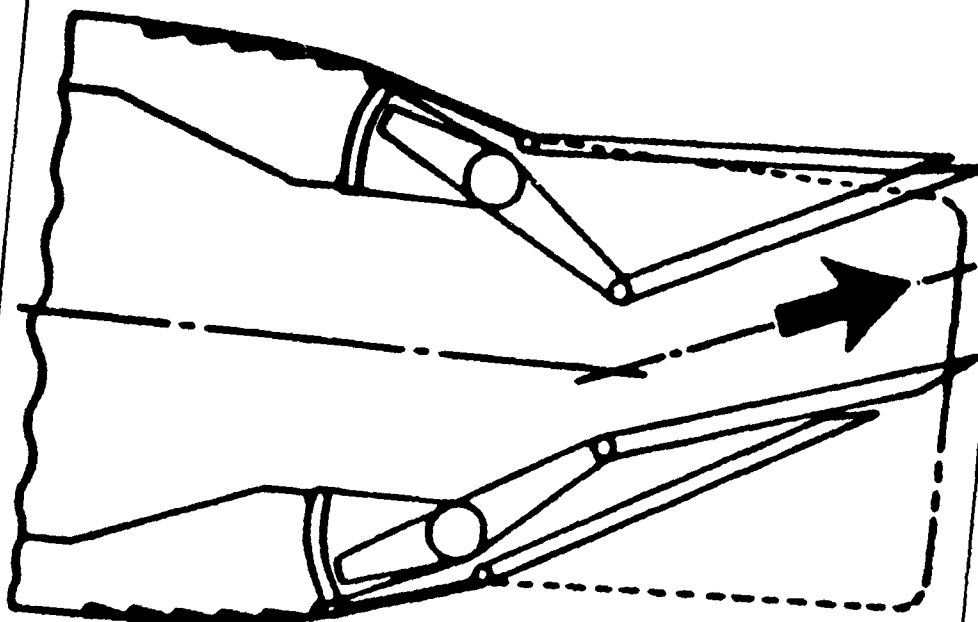
Fig. 2.5. Nozzle Design (15)

convergent flaps within the nozzle assembly. Unlike the rudders, the 2-D nozzles cannot produce direct side force (F_y). When deflected differentially, however, they assist in roll control, especially at higher Mach numbers (Figure (2.10)). The 2-D portion of the nozzle apparatus is used exclusively for in-flight maneuvering and takeoff (Figure 2.6).

The second essential part of the F-15/STOL's thrust vectoring capability comes from the rotating vane assembly positioned on the top and bottom of each engine exhaust. The vanes are primarily used during landing approach for precise control of velocity. These vanes can only be used when the 2-D nozzle is completely closed off. Since the vanes' movement is limited to ± 45 degs from vertical, the total available thrust during vane operation is $0.707 (\cos 45)$ of full military power (Figure 2.7). The rotating vanes' primary purpose during in-flight maneuvering is for rapid deceleration by thrust reversal (Figure 2.8). This study relies on small perturbations from equilibrium flight so as to maintain an accurate linear aircraft model. Therefore, the use of thrust reversal for rapid in-flight deceleration is not within the scope of this thesis.

The improvement in maneuverability is evident with the use of 2-D thrust vectoring nozzles (Figure 2.9). Preliminary wind tunnel data indicates a significant

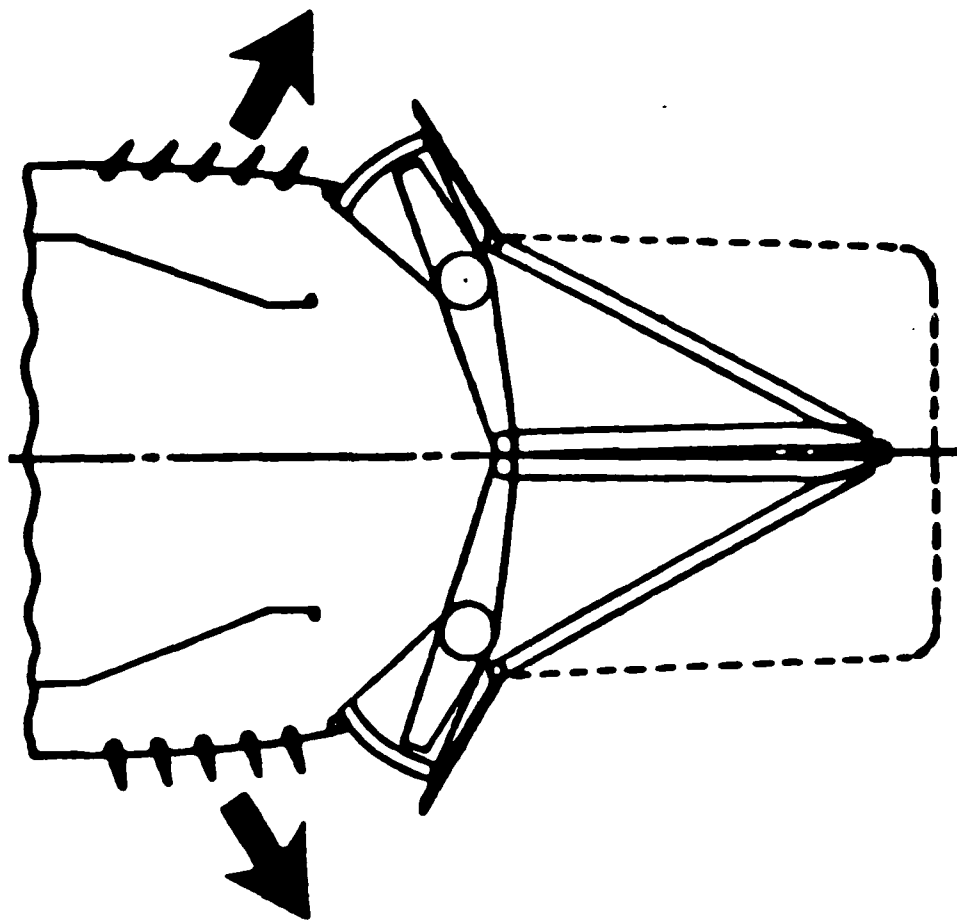
Primary Jet Vectoring



In-Flight Maneuvering/Takeoff

Fig. 2.6. Nozzle: Primary Jet Vectoring (15)

Rotating Vane Vectoring



Landing Approach Control

Fig. 2.7. Nozzle: Rotating Vane Vectoring (15)

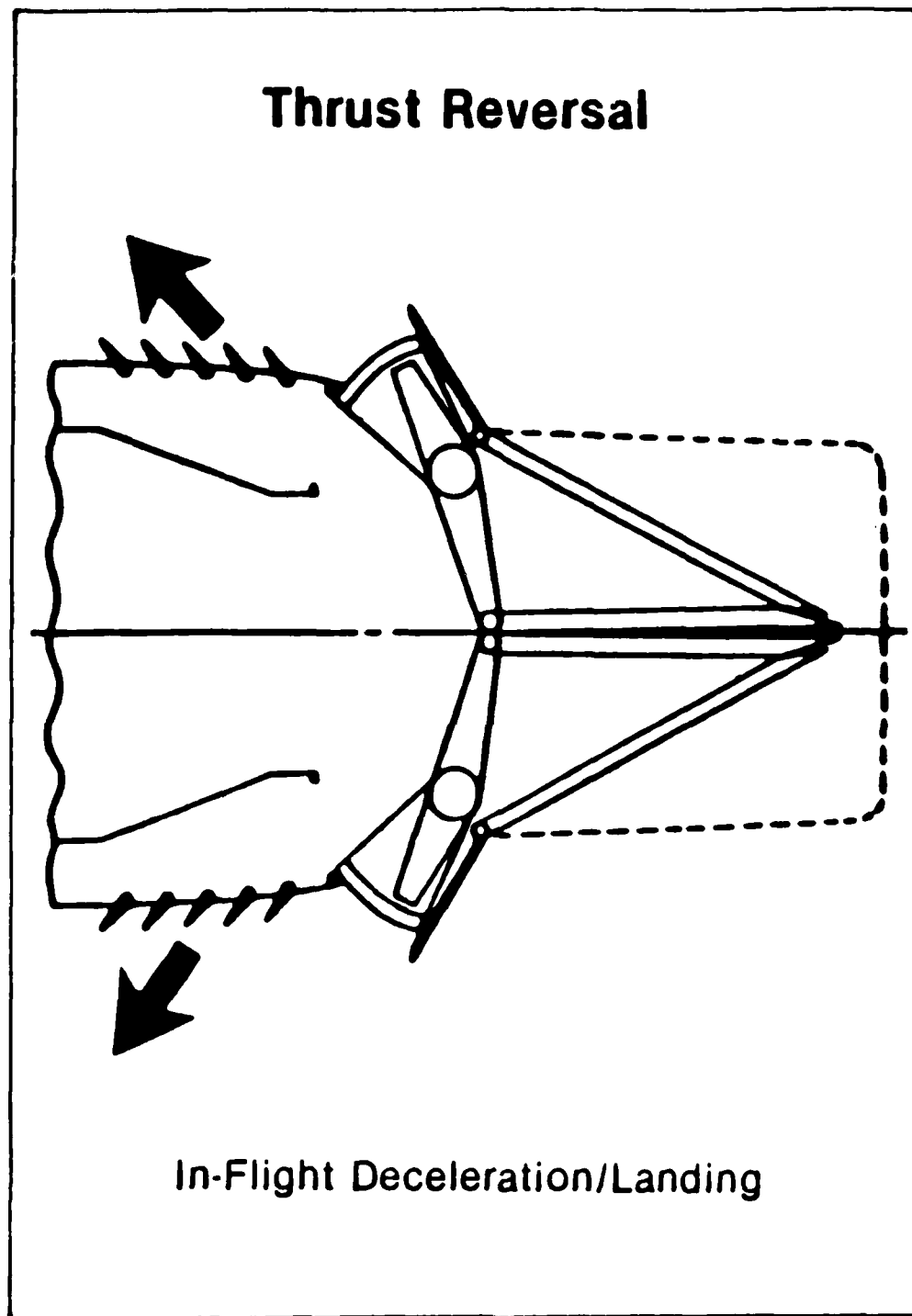


Fig. 2.8. Nozzle: Thrust Reversal (15)

Thrust Vectoring Enhances Agility

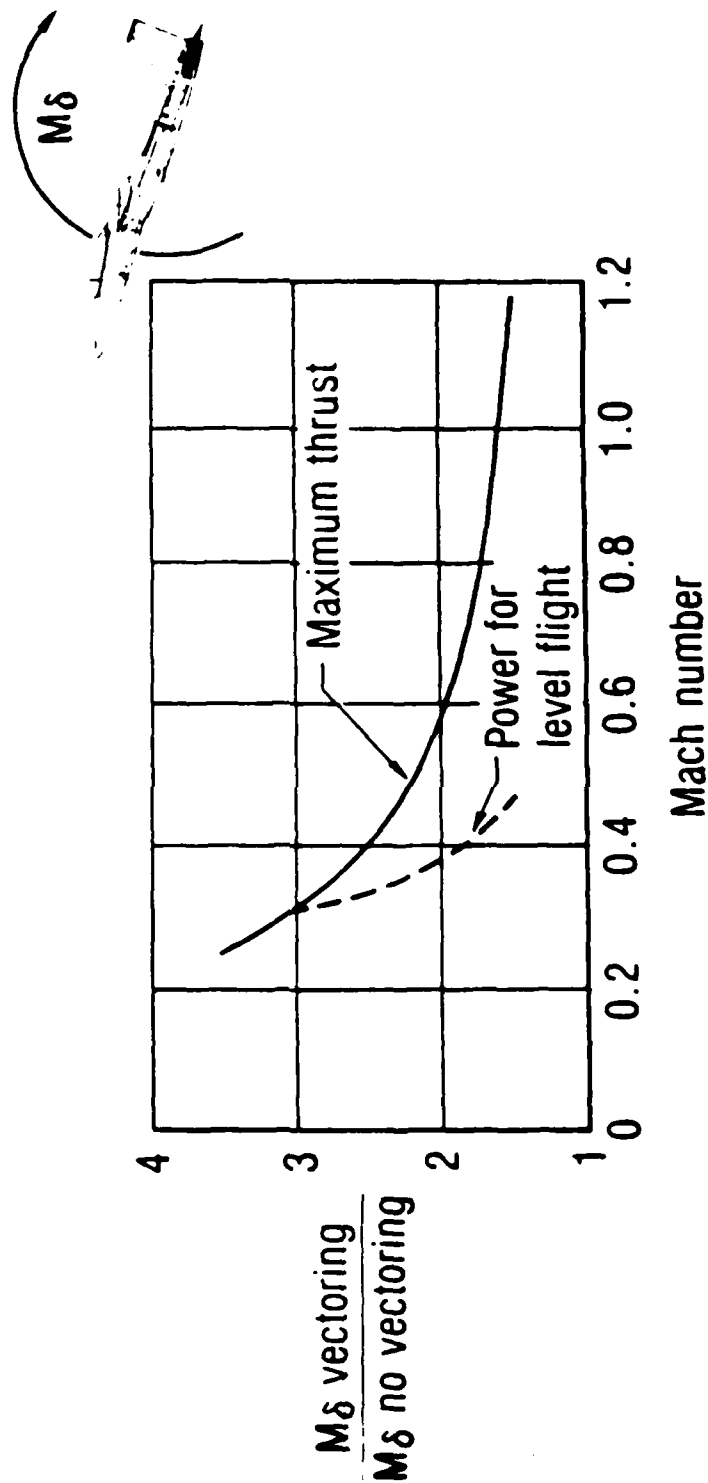


Fig. 2.9. Improved Pitching Moment (15)

improvement in pitching moment can be achieved by thrust vectoring, especially in the low Mach number region. Although not as significant, the differentially controlled nozzles also enhance this aircraft's roll performance (Figure 2.10). This minor improvement in roll authority is expected due to the nozzle's short moment arm from the aircraft's roll axis.

2.7 Summary

Preliminary wind tunnel data indicates that the F-15/STOL derivative fighter will provide significantly improved air combat maneuverability through the use of canard and thrust vectoring technology. As an added benefit, the additional control surfaces provide redundancy for the overall control law design. This redundancy is necessary for control reconfiguration in the event of failures through battle damage or other causes.

The next chapter develops the mathematical model used in designing the longitudinal control laws for the F-15/STOL.

Thrust Vectoring Enhances Roll Performance

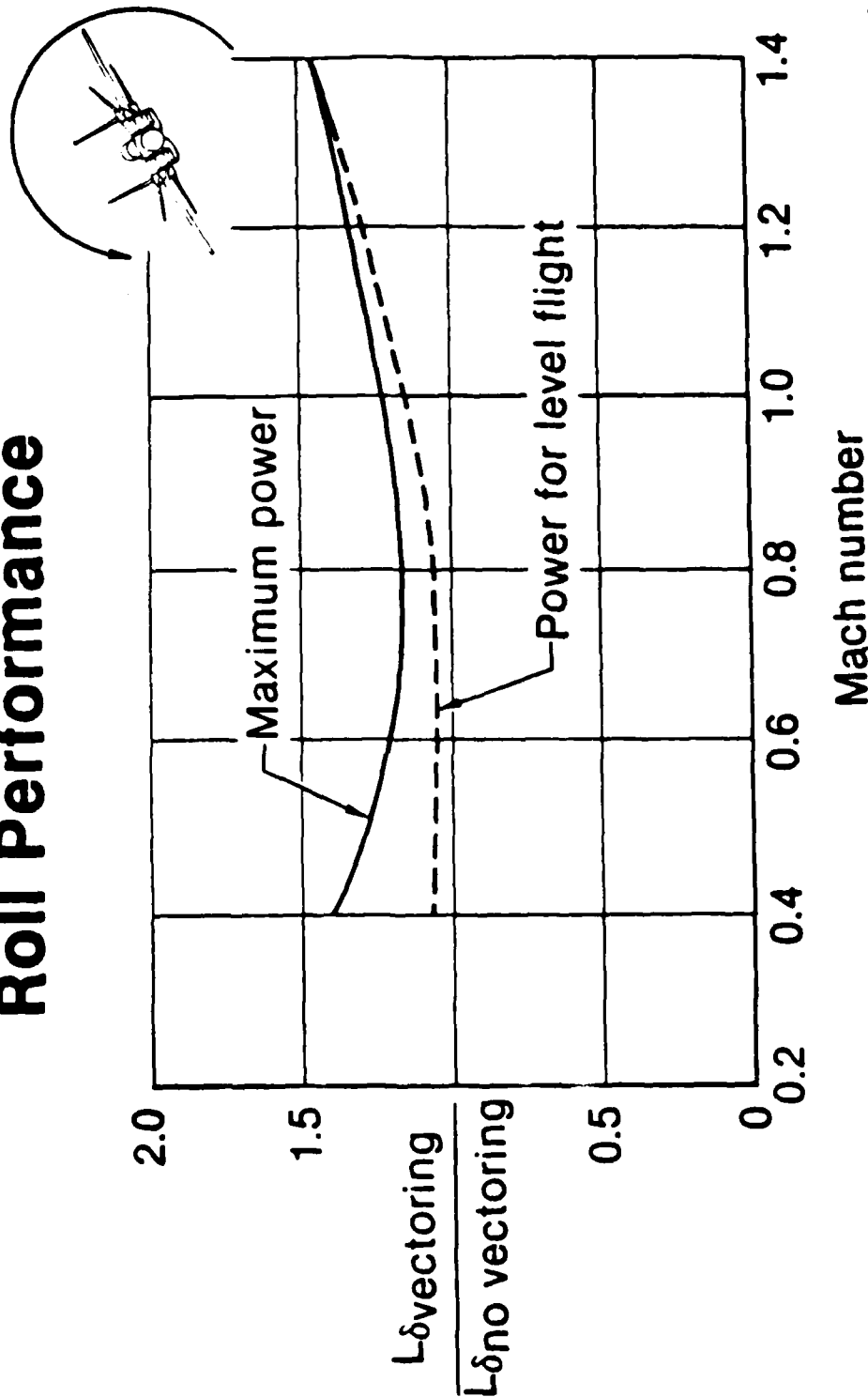


Fig. 2.10. Improved Rolling Moment (15)

III. The F-15/STOL Mathematical Model

3.1 Introduction

As with all engineering design procedures, the physical world must be suitably represented by a mathematical model that can be used for both the design and simulation of acceptable control laws. Whenever possible, linear approximations to the real world nonlinear aircraft are made which provides for a more direct design procedure. The Porter technique requires a linear, time-invariant model that is expressed in state-space form (18). The aircraft control law design problem is well suited to this particular mathematical representation.

The aircraft's motion is described by nonlinear force and moment equations that are linearized about an equilibrium trimmed condition. For small perturbations about that point, these equations relate the forces and moments generated by the control surfaces to the aircraft's linear and angular accelerations. These accelerations, or internal variables, are represented in the state-space model as states and state derivatives. The Porter design method provides for direct control over these internal variables through output feedback using the data available from onboard position and rate sensors.

This chapter describes the development of the mathematical model using data provided by McDonnell Aircraft Corporation (MCAIR) and the Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio. Besides the basic aircraft, models for the control surface actuator dynamics and onboard sensor dynamics are also presented. As with any model, there are limits and conditions to its range and validity. Therefore, the assumptions and limitations of the model are included in this chapter. In certain cases, which are described later, reasonable engineering approximations are used where necessary data is either insufficient or nonexistent.

3.2 Assumptions

This model incorporates the commonly used assumptions found in numerous reference texts that deal with aircraft models and equations of motion (3; 9; 21). Accurate results are achieved by limiting the simulation of the designed control laws by the bounds prescribed by the following assumptions:

1. The earth's surface is an inertial reference frame.
2. The atmosphere is fixed with respect to the earth.
3. The aircraft's mass is constant.
4. The aircraft is a rigid body.

5. The airstream surrounding the aircraft changes instantaneously following vehicle disturbances from equilibrium.

6. Aircraft aerodynamics are fixed for each equilibrium flight condition.

7. Linear perturbation equations are accurate for point designs.

8. Decoupling of the lateral and longitudinal equations of motion is acceptable.

Discussion of Assumptions.

1. Acceptance of the earth's surface as an inertial reference is valid for two reasons. First, the 16 sec duration of the maneuvers is negligible when compared with the earth's rotation rate. The reference frame's movement is insignificant during the simulated maneuver. Secondly, the onboard position and rate sensors (disregarding INS) are not sensitive enough to detect earth rotation rate or coriolis acceleration.

2. Modifications to the design program MULTI (Appendix A) allow for the addition of random wind gust effects during simulation. This thesis uses maneuvers at medium and high altitude to demonstrate the control law performance. During these maneuvers, wind gust effects are assumed negligible and not included in the simulation. The

assumption of a fixed atmosphere, therefore, is valid for this study.

3. The assumptions of constant mass and a rigid body are generally good for fighter aircraft. High fuel consumption rates along with fuel sloshing during extended air combat maneuvering can invalidate constant mass and fixed CG approximations. Since the maneuvers are simulated for only 16 secs, constant mass and fixed CG remain as good assumptions. Furthermore, flight outside of the transonic region generally diminishes wing flutter and other bending mode effects in fighter aircraft. As a result, rigid body assumptions remain valid for this model.

4. The assumption of instantaneously changing airflow allows for the elimination of the δ stability derivatives and greatly simplifies the aircraft modeling problem. This assumption is less valid in the transonic and supersonic regions where compressibility effects are significant. For the purposes of this study, the assumption of instantaneously changing airflow is made first, because unsteady airflow data is not available; and secondly, because it provides for an acceptable first approximation.

5. A common assumption which allows for linear perturbation equations to approximate the full nonlinear equations of motion for an aircraft is that the vehicle aerodynamics do not change for a fixed, trimmed flight condition (constant Mach number and altitude). In reality,

there is a slight change in velocity and altitude during the simulation maneuvers which results in small changes in the stability derivatives. These changes are considered insignificant for the purpose of this thesis and are therefore ignored.

6. This thesis studies the design of longitudinal controllers at four flight conditions within the aircraft's operating envelope (Table 3.5). The linearized equations of motion are used for the design at these specific points. To accomplish a design valid throughout the flight envelope, additional points would be selected for design in a similar manner with gain scheduling used between the points. With the linear point design assumption, the model is assumed valid for a small region around the equilibrium point.

7. The final assumption of a decoupled longitudinal mode requires the existence of a plane of symmetry (x-z plane), a flat earth, and no engine gyroscopic effects. All of these conditions are generally valid for the F-15/STOL model and therefore decoupling can safely be assumed.

The assumptions outlined above are commonly used in most aircraft models and are appropriate for the purposes of this study.

3.3 Model Derivation

At the time this study was undertaken, aerodynamic data for the F-15/STOL had been developed only to the point

of a cruise mode configuration. Cruise mode refers to trimmed flight conditions at minimum drag and is used for maximum range and endurance calculations. The model did not incorporate a thrust input and, consequently, any movement of the 2-D nozzles away from the body x-axis would decrease thrust at a fixed power setting. Two options were available at this point.

First, the standard four-state longitudinal model could be reduced to a three-state model by assuming a constant velocity and dropping the \dot{u} equation. With this model, a maximum of two inputs could be used to control two outputs. This method is used for the constant g pull-up maneuver. Thrust vectoring effects are retained by combining the nozzles and stabilators into a single control input while using the canards as a second independent input.

A second option would be to add a thrust input which would retain the four-state model but require a fixed nozzle because of nonlinearities that are discussed later in this chapter. This option is used for the remaining three maneuvers. A suitable engine model is derived using available data from a previous thesis modeling the X-29 experimental aircraft (6).

Equations of Motion. The equations of motion used in this thesis are the standard longitudinal force and

moment equations found in most aerodynamic reference texts (3; 9; 21). The body axis reference frame is used throughout this study for two reasons. First, the physical variables that the pilot attempts to control are sensed by both the pilot and the aircraft sensors in the body axis. These variables expressed in any other axis system lose their physical significance. Secondly, the simulation results are much easier to interpret when expressed in the body axis reference frame.

The aircraft's longitudinal motion can be expressed by three nonlinear equations; two force equations (x and z directions) and one moment equation about the y axis. Figure 3.1 depicts the body frame axis system for the F-15/STOL aircraft.

Summing the forces in the z direction gives:

$$F_{z_{cg}} = m(\dot{W} + pV - qU) - mg \cos \theta \cos \phi \quad (3-1)$$

Dividing both sides of the equation by m and rearranging to solve \dot{W} gives:

$$\dot{W} = F_{z_{cg}} / m - pV + qU + g \cos \theta \cos \phi \quad (3-2)$$

Numerous texts on aerodynamics develop the generalized perturbation equation from Equation (3-2) based on equilibrium flight conditions, i.e. $\phi = 0$, $p = q = 0$, and $\cos(\theta)$ approximately equal to unity (3; 9; 21). Only the results

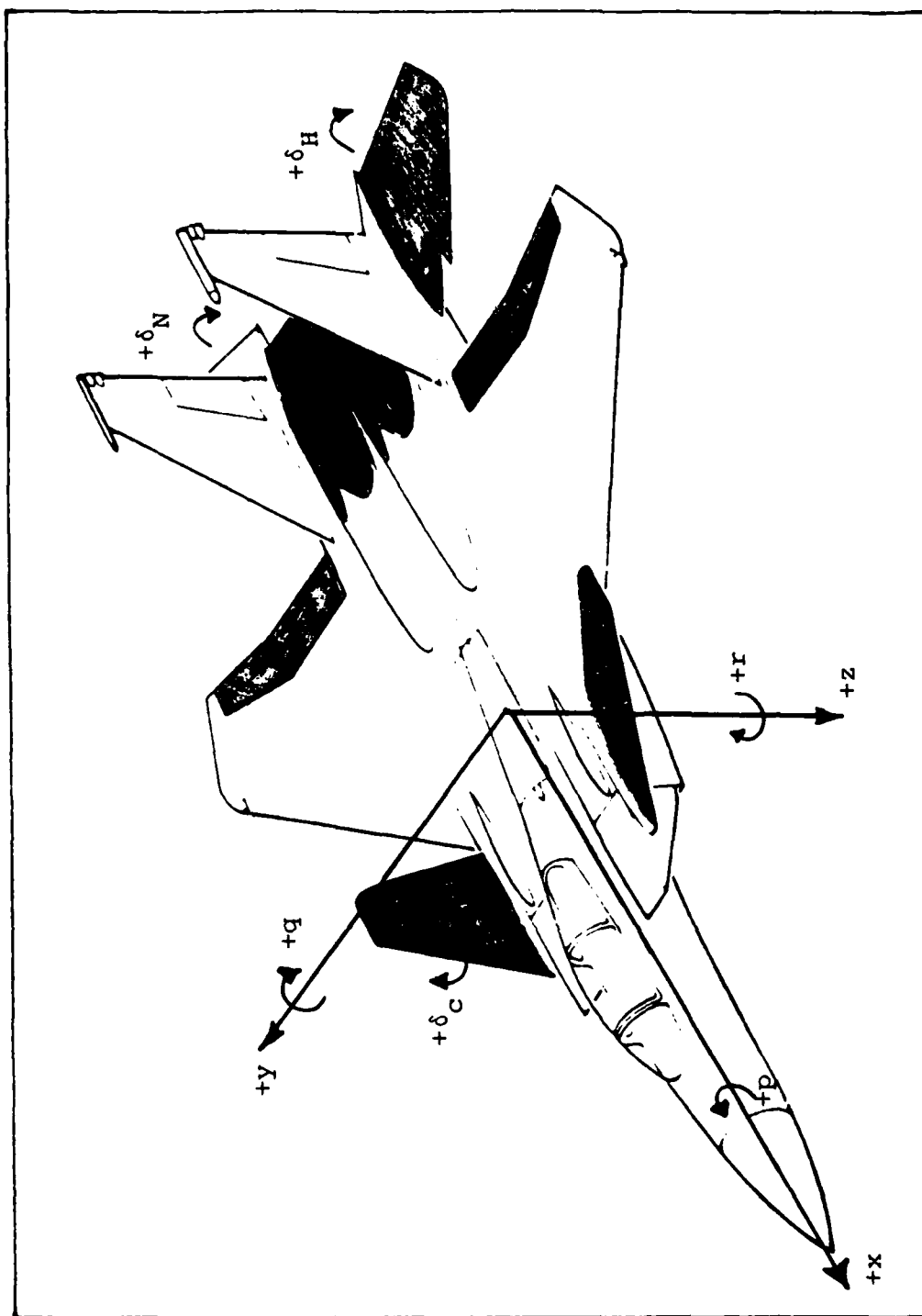


Fig. 3.1. Aircraft Body Axis System (15)

of this derivation are presented in this thesis. The generalized perturbation equation for vertical acceleration is given as:

$$\begin{aligned}\dot{w} = & (Z_{\theta})\theta + (Z_u)u + (Z_q)q + (Z_{\alpha})\alpha \\ & + (Z_{\delta_c})\delta_c + (Z_{\delta_H})\delta_H + (Z_{\delta_N})\delta_N\end{aligned}\quad (3-3)$$

Dividing by U, letting $\dot{\alpha} = \dot{w}/U$, and setting $z' = z/U$, Equation (3-3) becomes:

$$\begin{aligned}\dot{\alpha} = & (Z'_{\theta})\theta + (Z'_u)u + (Z'_q)q + (Z'_{\alpha})\alpha \\ & + (Z'_{\delta_c})\delta_c + (Z'_{\delta_H})\delta_H + (Z'_{\delta_N})\delta_N\end{aligned}\quad (3-4)$$

where

$$Z'_{\theta} = (-g \sin \theta_T) / U \quad (3-5a)$$

$$Z'_u = [(2) \bar{q} S / U^2 m] C_{z_u} \quad (3-5b)$$

$$Z'_q = \cos \alpha_T \quad (3-5c)$$

$$Z'_{\alpha} = [(57.3) \bar{q} S g / m U] C_{z_{\alpha}} \quad (3-5d)$$

$$Z'_{\delta_c} = [\bar{q} S g / m] C_{z_{\delta_c}} \quad (3-5e)$$

$$Z'_{\delta_H} = [\bar{q} S g / m] C_{z_{\delta_H}} \quad (3-5f)$$

$$\dot{z}'_{\delta_N} = [\bar{q}Sg/m]C_{z_{\delta_N}} \quad (3-5g)$$

Equations (3-5a) through (3-5g) are given in the MCAIR data package submitted for use in this study (15). For the F-15/STOL, C_{z_q} is equal to zero; therefore, Equation (3-5c) is presented in a simplified form.

Likewise, the force equation in the x direction is stated as follows:

$$F_{x_{cg}} = m(\dot{U} + qW - rV) + mg \sin \theta \quad (3-6)$$

rearranging yields

$$\dot{U} = F_{x_{cg}}/m - qW + rV - g \sin \theta \quad (3-7)$$

Using the same conditions of equilibrium flight as the z-force equation results in the following generalized perturbation equation:

$$\begin{aligned} \dot{u} = & (X'_\theta)\theta + (X'_u)u + (X'_q)q + (X'_\alpha)\alpha \\ & + (X'_{\delta_C})\delta_C + (X'_{\delta_H})\delta_H + (X'_{\alpha_N})\alpha_N \end{aligned} \quad (3-8)$$

where

$$X'_\theta = -g \cos \theta_T \quad (3-9a)$$

$$X'_u = [(2)g\bar{q}S/mU/C_{x_u}] \quad (3-9b)$$

$$\dot{x}'_q = -U \sin \alpha_T \quad (3-9c)$$

$$\dot{x}'_\alpha = [(57.3)\bar{q}Sg/m]C_{x_\alpha} \quad (3-9d)$$

$$\dot{x}'_{\delta_c} = [g\bar{q}S/m]C_{x_{\delta_c}} \quad (3-9e)$$

$$\dot{x}'_{\delta_H} = [g\bar{q}S/m]C_{x_{\delta_H}} \quad (3-9f)$$

$$\dot{x}'_{\delta_N} = [g\bar{q}S/m]C_{x_{\delta_N}} \quad (3-9g)$$

Equations (3-9a) through (3-9g) are obtained from Reference (15). In addition, C_{x_q} for this model is equal to zero and therefore Equation (3-9c) is given in the simplified form.

The third equation defines the pitching moment about the y-axis, M_y . The general moment equation is written as:

$$M_y = \dot{q} I_{yy} + pr(I_{xx} - I_{zz}) - (r^2 - p^2)I_{xz} \quad (3-10)$$

Because longitudinal motion is confined to the x-z plane with zero bank angle, roll rate, and yaw rate, the moment equation simplifies to:

$$\dot{q} = M_y / I_{yy} \quad (3-11)$$

In a similar development of the general perturbation equation from Equation (3-11), the following result is written:

$$\begin{aligned}\dot{q} = & (M'_u)u + (M'_q)q + (M'_\alpha)\alpha + (M'_{\delta_c})\delta_c \\ & + (M'_{\delta_H})\delta_H + (M'_{\delta_N})\delta_N\end{aligned}\quad (3-12)$$

where

$$M'_u = [(2)\bar{q}Sc/I_{yy} U] C_{m_u} \quad (3-13a)$$

$$M'_q = [(57.3)\bar{q}Sc^2/(2)UI_{yy}] C_{m_q} \quad (3-13b)$$

$$M'_\alpha = [(57.3)\bar{q}Sc/I_{yy}] C_{m_\alpha} \quad (3-13c)$$

$$M'_{\delta_c} = [\bar{q}Sc/I_{yy}] C_{m_{\delta_c}} \quad (3-13d)$$

$$M'_{\delta_H} = [\bar{q}Sc/I_{yy}] C_{m_{\delta_H}} \quad (3-13e)$$

$$M'_{\delta_N} = [\bar{q}Sc/I_{yy}] C_{m_{\delta_N}} \quad (3-13f)$$

The coefficient for θ , M'_θ , is proportional to M'_α and therefore equal to zero from assumption 5, Sec. 3-2. Equations (3-13a) through (3-13f) are obtained from Reference (15).

The final equation in the four-state model is simply the kinematic relationship,

$$\dot{\theta} = q \quad (3-14)$$

Pitch rate is defined as the time-rate-of-change of the pitch angle θ when bank angle ϕ is equal to zero. This equation is necessary to form a square plant matrix in the state-space representation.

It is important to keep in mind that there are still only three degrees of freedom represented by this four-state model. Hence, there is a physical limitation of at most three independent inputs that can be used to control the four physical variables, if a unique solution is to be determined. Any additional inputs must be either weighed by a desired optimality criterion or physically combined with one of the original three inputs.

State-Space Form. The four equations of motion developed in the previous section can be represented in the familiar state-space form as:

$$\dot{\underline{x}} = \underline{A}\underline{x} + \underline{B}\underline{u} \quad (3-15a)$$

$$\underline{y} = \underline{C}\underline{x} \quad (3-15b)$$

where

\underline{A} = square plant matrix ($n \times n$)

\underline{B} = control input matrix ($n \times m$)

\underline{C} = output matrix ($l \times n$)

with \underline{x} = state vector, \underline{u} = control input vector, and \underline{y} = output vector. Writing out Equations (3-4), (3-8), (3-12), and (3-14) in the form above gives the following state

$$\begin{bmatrix} \dot{\theta} \\ \dot{u} \\ \dot{q} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ X'_{\theta} & X'_u & X'_q & X'_{\alpha} \\ 0 & M'_u & M'_q & M'_{\alpha} \\ Z'_{\theta} & Z'_u & Z'_q & Z'_{\alpha} \end{bmatrix} \begin{bmatrix} \theta \\ u \\ q \\ \alpha \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ X'_{\delta_C} & X'_{\delta_H} & X'_{\delta_N} \\ M'_{\delta_C} & M'_{\delta_H} & M'_{\delta_N} \\ Z'_{\delta_C} & Z'_{\delta_H} & Z'_{\delta_N} \end{bmatrix} \begin{bmatrix} \delta_C \\ \delta_H \\ \delta_N \end{bmatrix} \quad (3-16a)$$

$$\begin{bmatrix} \theta \\ u \\ \gamma \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \theta \\ u \\ q \\ \alpha \end{bmatrix} \quad (3-16b)$$

The Porter method, using output feedback, allows the designer to freely choose the y vector in order to achieve a desired response. The output vector from Equation (3-16b)

is chosen for ease in modeling the simulation maneuvers. Other theses have used the \underline{C} matrix to form such output variables as normal cockpit acceleration A_{n_p} , which is easily sensed and very desirable to control (2).

The model described in Equations (3-16a) and (3-16b) is modified by fixing the nozzle at its trimmed angle (approximately zero degrees) and replacing it with a thrust input. Since the thrust vector is always aligned with the aircraft's body x-axis, changes in thrust do not contribute to either the F_z or M_y equations. Consequently, M'_{δ_T} and Z'_{δ_T} are both set to zero. With the addition of adjustable thrust, the \underline{B} matrix for this model is now given as:

$$\underline{B}\underline{u} = \begin{bmatrix} 0 & 0 & 0 \\ X'_{\delta_C} & X'_{\delta_H} & X'_{\delta_T} \\ M'_{\delta_C} & M'_{\delta_H} & 0 \\ Z'_{\delta_C} & Z'_{\delta_H} & 0 \end{bmatrix} \begin{bmatrix} \delta_C \\ \delta_H \\ \delta_T \end{bmatrix} \quad (3-17)$$

where δ_T is now expressed as a throttle angle ratio.

This model is used for the simulation of three longitudinal maneuvers: pitch pointing, vertical translation and direct climb. These maneuvers are described in detail later in the Chapter. Table 3.1 lists the open-loop eigenvalues for the continuous time plant at each of

TABLE 3.1
OPEN-LOOP PLANT EIGENVALUES

Flt Cond	Four-State Model	Three-State Model
0.3 Mach FL 200	-.1494+01 + J0	N/A
	.2811+00 + J0	
	-.1326-01 + J.5204-01	
	-.1326-01 - J.5204-01	
0.9 Mach FL 200	-.5049+01 + J0	-.5048+01 + J0
	.1551+01 + J0	.1548+01 + J0
	-.9267-02 + J.3692-01	.1024-02 + J0
	-.9267-02 - J.3692-01	-
1.4 Mach FL 200	-.3700+01 + J.7084+01	-.3700+01 + J.7084+01
	-.3700+01 - J.7084+01	-.3700+01 - J.7084+01
	-.8492-02 + J.3659-01	-.5938-04 + J0
	-.8492-02 - J.3659-01	-
2.0 Mach FL 400	-.4866+01 + J.2917+01	-.4865+01 + J.2917+01
	-.4866+01 - J.2917+01	-.4865+01 - J.2917+01
	.3649-01 + J0	.3364-04 + J0
	-.4933-01 + J0	-

the four flight conditions. The roots clearly indicate the static instability of the uncontrolled plant.

As mentioned earlier, the potential benefits of thrust vectoring cannot be explored in the simulation when replacing the nozzle with the throttle input. For this reason, the three-state model is also used in the simulations since it retains the vectored nozzle effects by combining the nozzle and stabilator into a single input.

Reduced Order Model. Assuming a constant velocity, i.e. enough thrust to force the perturbation velocity u to zero throughout the time of simulation, allows for a

simplification of the four-state model. The \dot{u} equation along with its associated state variable u , can be eliminated, resulting in the reduced order model (short-period approximation):

$$\begin{bmatrix} \dot{\theta} \\ \dot{q} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & M'_q & M'_\alpha \\ Z'_\theta & Z'_q & Z'_\alpha \end{bmatrix} \begin{bmatrix} \theta \\ q \\ \alpha \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ M'_{\delta_C} & M'_{\delta_{HN}} \\ Z'_{\delta_C} & Z'_{\delta_{HN}} \end{bmatrix} \begin{bmatrix} \delta_C \\ \delta_{HN} \end{bmatrix} \quad (3-18a)$$

with the output relationship stated as:

$$\begin{bmatrix} \theta \\ \alpha \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta \\ q \\ \alpha \end{bmatrix} \quad (3-18b)$$

where the eigenvalues for the open-loop plant are listed in Table 3.1. This model is used for the constant g pull-up maneuver. The results demonstrate the important contribution that thrust vectoring can make to the vehicle's pitching moment (Chapter V).

The u control vector in Equation (3-18a) is composed of the canard and a new control input, δ_{HN} . This new input is formed by combining the stabilator and nozzle. Their similar effects on pitching moment, due to their

position aft of the aircraft's CG, makes them the logical choice for combination. Two physical problems occur when inputs are combined. First, suitable actuator dynamics must be modeled. The solution to this is described later in the chapter. Secondly, input saturation can occur since both inputs have different deflection limits. This problem is easily solved by weighing the two control derivatives in such a manner that both inputs saturate simultaneously. In the case of the stabilator and nozzle, the deflection limits are -29 to +15 degs and -/+20 degs respectively. Since each input is trimmed at a specific deflection angle for each equilibrium point, the effective deflection limits about the trimmed value must be computed.

As an example, at 0.3 Mach and FL200, the trimmed values for stabilator and nozzle are +4.555 and 0 degs respectively. Given the original limits, new deflection limits for the stabilator about its trimmed value are -33.55 and +10.44 degs. The nozzle limits remain unchanged. Since the stabilator's negative limit is greater than the nozzle's limit, the nozzle would be driven into saturation if an unweighted combination were used. The new control derivatives are formed in a weighted combination as

$$M'_{\delta_{HN}} = M'_{\delta_H} + [(20)/(33.55)]M'_{\delta_N} \quad (3-19a)$$

$$Z'_{\delta_{HN}} = Z'_{\delta_H} + [(20)/(33.55)]Z'_{\delta_N} \quad (3-19b)$$

for this particular flight condition. Now, by using the stabilator deflection limits for this new control input, the nozzle saturates simultaneously with the stabilator at the negative deflection limit. For positive deflections, however, the nozzle never reaches saturation. Specifically, for this flight condition, a maximum control input of +10.44 degs results in a nozzle deflection of 6.22 degs or approximately 31 percent of its allowable travel (10.44/33.55). A modification to MULTI's simulation capability would eliminate this deficiency and is included as a recommendation. This input combination calculation is made for each of the three flight conditions where the constant g pull-up is simulated.

Other theses have controlled cockpit acceleration, A_{n_p} , by including it as a state within the plant matrix \underline{A} (2). When employing the Porter design technique, this method results in two transmission zeros at the origin (Appendix B). In an effort to avoid this problem, this thesis controls acceleration in a more indirect fashion. From dynamics, A_{n_p} is defined as:

$$A_{n_p} = \dot{\gamma}U + \dot{q}\ell_x \quad (3-20)$$

where ℓ_x is the distance between the cockpit and the aircraft's CG. Since $\dot{\gamma} = q - \dot{\alpha}$, A_{n_p} can be expressed as a linear combination of states and state derivatives.

Furthermore, in the steady-state, $\dot{\alpha}$ and \dot{q} go to zero, leaving A_{n_p} proportional to q . When A_{n_p} is expressed in g's, the relationship simplifies to

$$A_{n_p} = \left[\frac{U}{1845} \right] q \quad (3-21)$$

where $1845 = 57.3 \text{ deg/rad} \times 32.2 \text{ ft/sec}^2$. The desired g force for the maneuver is commanded by ramping θ (the slope = q) at the appropriate value.

From the model given in Equation (3-18), the equations for $\dot{\alpha}$ and \dot{q} are:

$$\dot{\alpha} = (Z'_\theta)\theta + (Z'_q)q + (Z'_\alpha)\alpha + (Z'_{\delta_c})\delta_c + (Z'_{\delta_{HN}})\delta_{HN} \quad (3-22a)$$

$$\dot{q} = (M'_q)q + (M'_\alpha)\alpha + (M'_{\delta_c})\delta_c + (M'_{\delta_{HN}})\delta_{HN} \quad (3-22b)$$

Substituting into Equation (3-20) and rearranging gives:

$$\begin{aligned} A_{n_p} = & (-Z'_\theta U)\theta + (M'_q \ell_x - Z'_\alpha U + U)q \\ & + (M'_\alpha \ell_x - Z'_\alpha U)\alpha + (M'_{\delta_c} \ell_x - Z'_{\delta_c} U)\delta_c \\ & + (M'_{\delta_{HN}} \ell_x - Z'_{\delta_{HN}} U)\delta_{HN} \end{aligned} \quad (3-23)$$

Through a modification to MULTI which allows for the plotting of a linear combination of both the states and the control inputs, A_{n_p} is directly available as an output (Appendix A).

Actuator Dynamics. Transfer functions representing the actuator dynamics for the canard, stabilator, and 2-D nozzle are provided by MCAIR; however, approximations are made so that the data is compatible with the MULTI simulation routine. The MCAIR stabilator and canard actuator transfer function is represented in the s-domain as:

$$\frac{\delta_c}{e_{\delta_c}} = \frac{1}{\left(\frac{s}{30.62} + 1\right) \left[\left(\frac{s}{272.9}\right)^2 + \frac{(2)(0.508)}{(272.9)} s + 1\right]} \quad (3-24)$$

This is reduced to an equivalent second-order model because of a current limitation on actuator model size in MULTI. The reduced order model is formed by eliminating one of the poles of the quadratic which gives:

$$\frac{\delta_c}{e_{\delta_c}} = \frac{1}{\left(\frac{s}{30.62} + 1\right) \left(\frac{s}{272.9} + 1\right)} \quad (3-25)$$

The time response plots for the two models given a step input is shown in Figure 3.2. The results show identical time response characteristics with the figures of merit listed in Table 3.2. By letting $X_1 = \delta_c$, $X_2 = \dot{X}_1$, and $y = X_1$, the state-space form for the canard and stabilator reduced order model is given as:

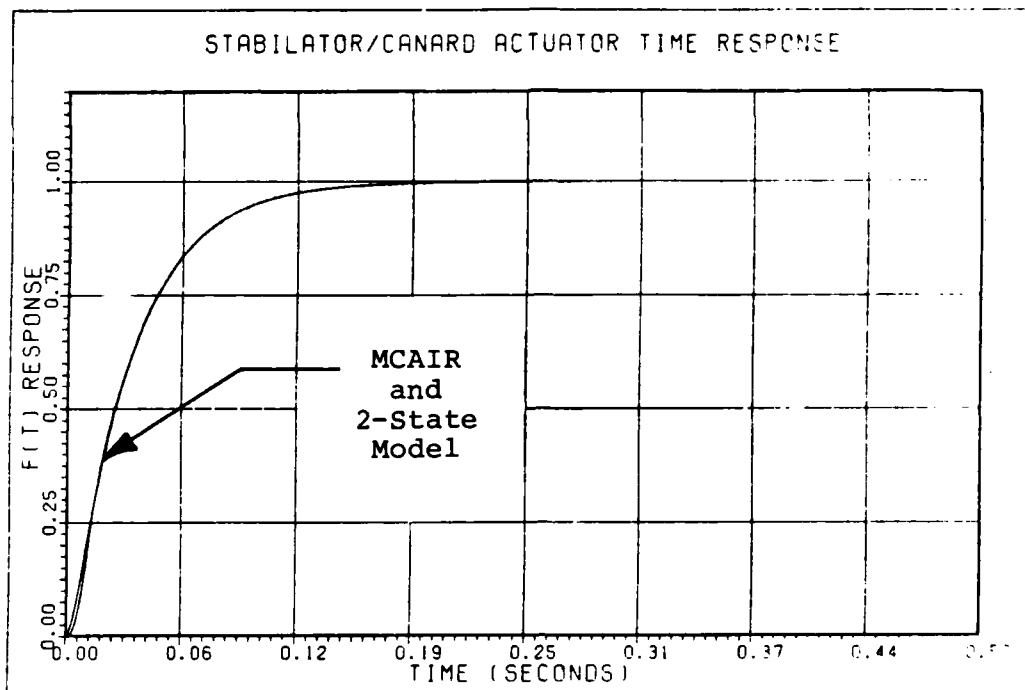


Fig. 3.2. Stabilator/Canard Actuator Time Response

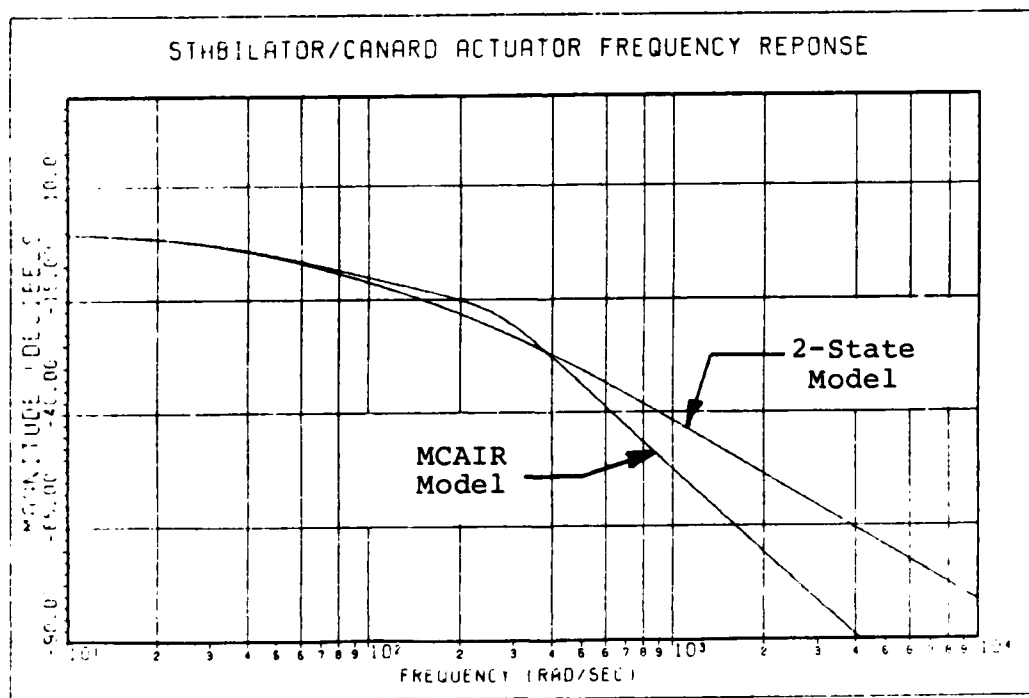


Fig. 3.3. Stabilator/Canard Actuator Frequency Response

TABLE 3.2
ACTUATOR MODEL FOM* COMPARISON

Actuators	FOM	MCAIR Model	Reduced-Order Model
Canard & Stabilator	(time-sec)		
	rise =	0.7041-01	0.7251-01
	duplication =	Very Large	Very Large
	peak =	Very Large	Very Large
4-State Plant	settling =	0.1313	0.1316
	(value)		
	peak =	1.0	1.0
	final =	1.0	1.0
Stabilator-Nozzle	(time-sec)		
	rise =	0.8195-01	0.7251-01
	duplication =	Very Large	Very Large
	peak =	Very Large	Very Large
3-State Plant	settling =	0.1487	0.1316
	(value)		
	peak =	1.0	1.0
	final =	1.0	1.0

* FOM = figures of merit.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -8356.2 & -303.5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 8356.2 \end{bmatrix} e_{\delta_c}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (3-26)$$

Figure 3.3 compares the MCAIR model against the reduced order model in the frequency domain. In the bandwidth of interest, the two models again show good agreement.

The MCAIR model for the nozzle actuator is a simple first order transfer function with a single pole at $s = -24$. In the three-state aircraft model used for the constant g maneuver, the stabilator and nozzle are combined into a single input. This input scheme requires a combined actuator model that is formed by adding the two transfer functions that operate in parallel. The resulting model is described as:

$$\frac{\delta_{HN}}{e_{\delta_{HN}}} = \frac{(12)(s+26.74)(s+625)}{(s+30.62)(s+272.9)(s+24)} \quad (3-27)$$

Again, because of the second-order limitation on MULTI's actuator models, the stabilator model given in Equation (3-26) is used for the stabilator-nozzle combined input. Figures 3.4 and 3.5 compare the time and frequency responses respectively of the models given by Equations (3-26) and (3-27). Table 3.2 compares the figures of merit. The similarity of the responses justifies the approximation used in this case.

As mentioned earlier, no thrust input is included in the MCAIR model submitted for this study. Consequently, engine data obtained from a previous thesis using the X-29 aircraft model is modified for use with the F-15/STOL (6). Throttle actuator dynamics are modeled as a first-order transfer function with a 0.05 sec time constant:

$$\frac{\delta_T}{e_{\delta_T}} = \frac{20}{s+20} \quad (3-28)$$

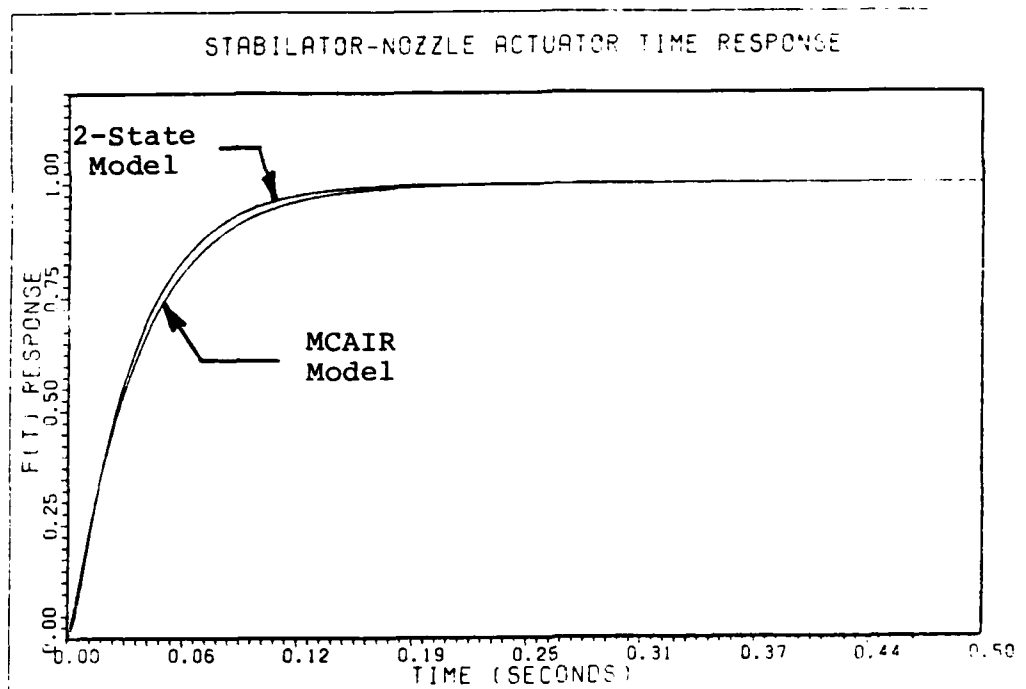


Fig. 3.4. Stabilator-Nozzle Actuator Time Response

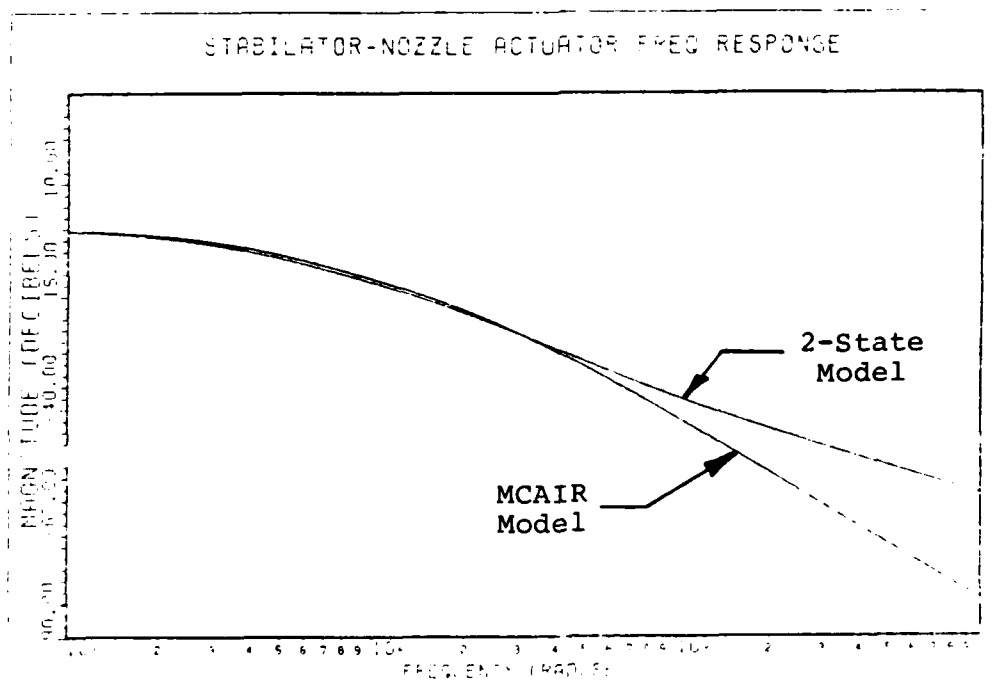


Fig. 3.5. Stabilator-Nozzle Actuator Frequency Response

Sensor Dynamics. The desired outputs from the continuous time plant are fed back and summed with the pilot input vector (\underline{v}) to form an error signal which is digitally sampled and input to the controller (Figure 3.6). The outputs for the four-state model are θ , u , and γ . For the short-period approximation model, they are θ and α . A measurement matrix is used in the design process for both of these models to give $\underline{F}_2 \underline{B}_2$ full rank and allow for the existence of $(\underline{F}_2 \underline{B}_2)^{-1}$ (Chapter IV). This results in an additional feedback of pitch rate q . The current version of MULTI does not allow for sensor dynamics affecting the variables fed back by the measurement matrix. This improvement is recommended for future work. Sensor dynamics, however, are included for the remaining output variables.

The sensor model includes both the sensor dynamics and an aliasing filter tuned for the sampling rate of 40 Hz. For all of the angle outputs (theta, gamma, and alpha), the second-order sensor model is given as:

$$\frac{y_{fb}}{y_{out}} = \frac{2926}{(s+14)(s+209)} \quad (3-29)$$

This model is a compromise between the first-order AOA model ($s = -14$) and the second-order aliasing filter ($\omega_n = -209$). Figures 3.7 and 3.8 compare the time and frequency responses respectively for the MCAIR model vs. the model given in Equation (3-29). Again, the approximation is

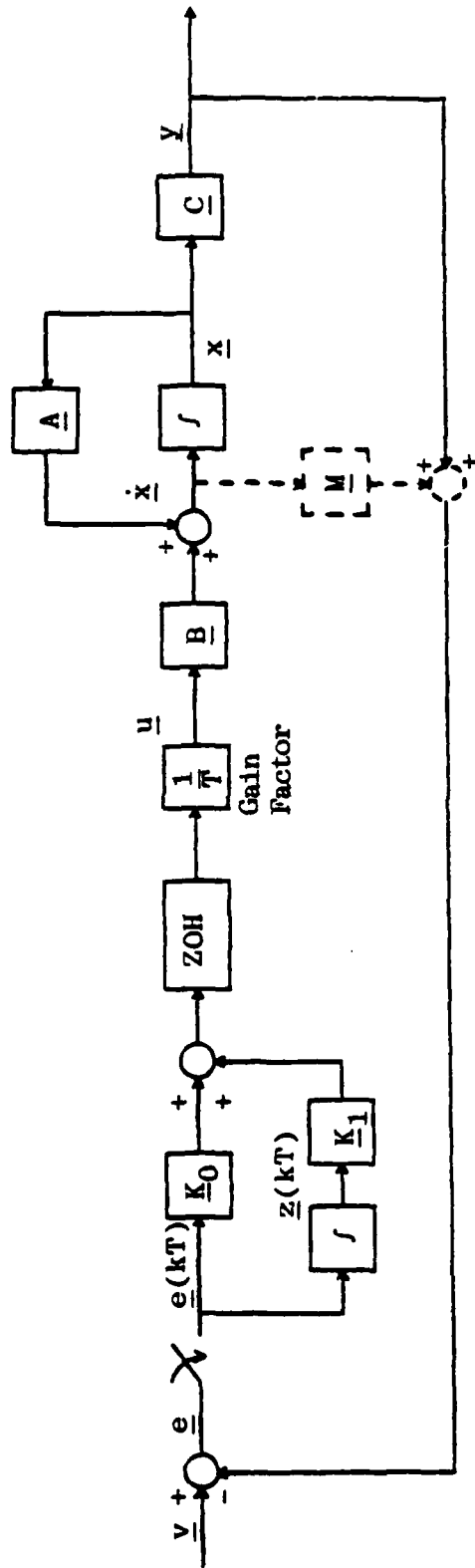


Fig. 3.6. Digital Block Diagram Configuration (2)

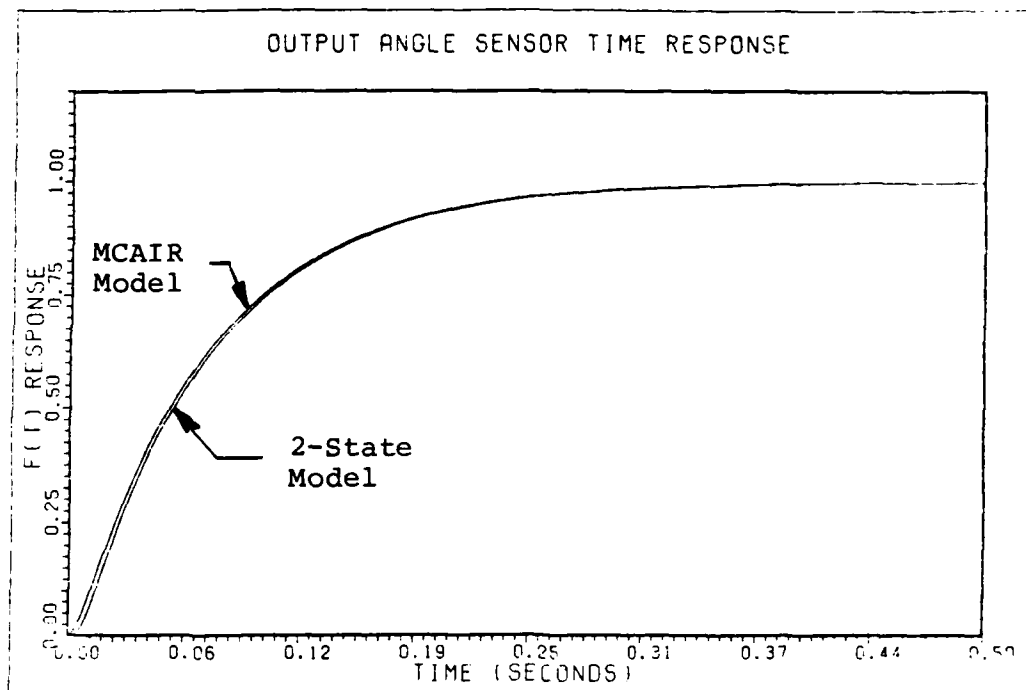


Fig. 3.7. Output Angle Sensor Time Response

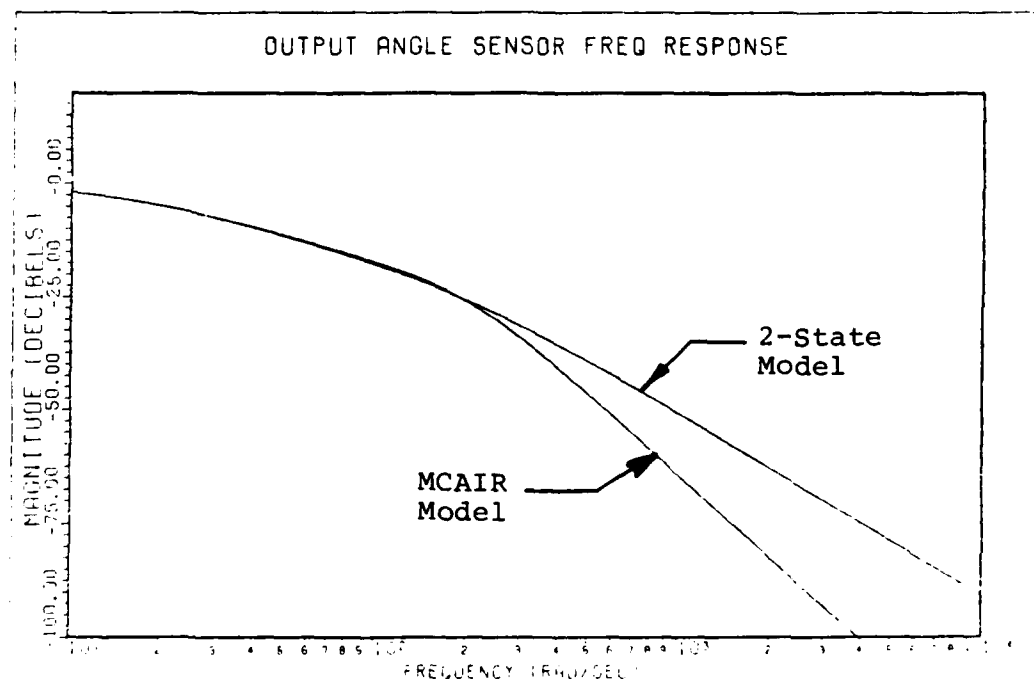


Fig. 3.8. Output Angle Sensor Frequency Response

acceptable for the purpose of this study. Table 3.3 compares the figures of merit between these models for a step input.

TABLE 3.3
SENSOR MODEL FOM* COMPARISON

Sensors	FOM	MCAIR Model	Reduced-Order Model
	(time-sec)		
Output	rise = 0.156559		0.157405
Angle	duplication = Very Large		Very Large
Sensor	peak = Very Large		Very Large
	settling = 0.286553		0.284383
	(value)		
3 & 4-State	peak = 1.0		1.0
Plant	final = 1.0		1.0

* FOM = figures of merit.

The MCAIR velocity sensor model is composed of an aliasing filter and a zero-order hold model. Since their version is second-order, it can be implemented directly into MULTI without an approximation. The velocity sensor model is given as:

$$\frac{y_{fb}}{y_{out}} = \frac{1200}{(s+30)(s+40)} \quad (3-30)$$

3.4 Limitations

All linear models must operate within prescribed limits that prevent invalidation of the model. Output

results must then be compared against these limitations to ensure that the linearity assumptions remain valid. For the F-15/STOL, pitch angle limited to ± 20 degs, AOA limited to ± 5 degs, and velocity limited to ± 5 ft/sec should provide accurate results. In addition, physical limitations on the control inputs such as maximum and minimum deflection limits along with maximum deflection rates must also be observed. These limitations are listed in Table 3.4.

TABLE 3.4
CONTROL INPUT DEFLECTION/RATE LIMITS

Input	Deflection Limits	Rate Limit
Canard (δ_C)	-35 to +15 degs	23 deg/sec
Stabilator (δ_H)	-29 to +15 degs	46 deg/sec
Nozzle (δ_N)	-20 to +20 degs	30 deg/sec
Throttle (δ_T)	-.2 to +1.0*	N/A

* throttle angle ratio (no units).

It is important to point out that the control inputs also have linear operating ranges that are well below the maximum values shown in Table 3.4. This problem, along with its implications, are discussed in detail in Chapter VI.

Currently, MULTI does not include rate limits as part of the simulation. Maximum control input rates (\dot{u}) must be evaluated from the output responses. In many cases, exceeding the maximum rates cannot be prevented without a "built in" rate limit as part of the simulation. This modification to MULTI is another recommendation for future work.

3.5 Model Nonlinearities

Two problems are encountered in this study that result in plant nonlinearities: "sign swapping" on the control derivatives of the \dot{u} equation and the time-varying B matrix resulting from models that use both independent thrust and nozzle inputs.

The "sign swapping" problem can be present in any aircraft model that uses aerodynamic surfaces as inputs to the \dot{u} equation. Since the force contributed by any control input is equal to the product of its control derivative and the input deflection, control surfaces can "appear" to produce thrust. Either a negative deflection with a negative control derivative or a positive deflection with a positive control derivative will result in a positive force in the +x direction. Since the control surfaces are frequently trimmed at non-zero AOA's, small perturbations about this trim point give accurate results. If the magnitude of the deflection causes the surface to transit

through zero AOA, however, the relationship is no longer valid since further deflection increases drag thereby increasing force in the -x direction.

This problem is solved by incrementally testing the control surface AOA during the simulation and switching the sign of its derivative when zero AOA is traversed. This becomes a nonlinear simulation since B matrix elements are no longer constant. Further explanation along with examples of both a linear and nonlinear simulation are given in Chapter IV.

Models that use both an independent nozzle and thrust input require a time-varying B matrix since the equations relating the force to the control input are given as:

$$F_{x \delta_T} = T \cos \delta_N \quad (3-31a)$$

$$F_{z \delta_T} = T \sin \delta_N \quad (3-31b)$$

Using the small angle approximation, the right side of Equation (31a) is simply equal to thrust, T. This can be represented by a constant control derivative times the throttle input. In the z direction, however, the force is now a product of the thrust and the nozzle deflection angle:

$$F_{z \delta_T} \approx T \delta_N \quad (3-32)$$

This model can be simulated by incorporating thrust as a state variable and then incrementally updating the B matrix control derivatives as thrust changes during the simulation.

The problem with the implementation of this scheme is the limitation imposed by the available outputs that can be fed back. The new state variable T (thrust) must be fed back since it is the only variable remaining (q is already fed back through the measurement matrix). Since T is directly proportional to γ , it is not an independent input for any particular maneuver. Consequently, the non-linear equations must be solved for the thrust required to perform the maneuver before the simulation is attempted. It is not clear just how to proceed from this point and, therefore, due to time constraints, this problem is left for future research.

Because of this problem, the four-state model in this study uses a fixed nozzle with a variable thrust input. Results from this research, therefore, form a basis that future studies can compare with when assessing the performance benefits of vectored thrust.

3.6 Simulation Maneuvers

The control laws developed in this thesis are designed around four longitudinal maneuvers and simulated at four flight conditions. Table 3.5 summarizes this data.

TABLE 3.5
SIMULATION MANEUVERS AND FLIGHT CONDITIONS

Flight Condition	Maneuver			
	Pitch Pointing	Vertical Translation	Direct Climb	Constant g Pull-Up
0.3 Mach FL 200	Yes	Yes	Yes	No
0.9 Mach FL 200	Yes	Yes	Yes	Yes
1.4 Mach FL 200	Yes	Yes	Yes	Yes
2.0 Mach FL 400	Yes	Yes	Yes	Yes

The simulation maneuvers demonstrate the decoupling of the output variables by commanding each output to a specified value. Each maneuver is performed about the equilibrium point for each flight condition. The four selected maneuvers are described as follows:

1. Pitch Pointing. This maneuver demonstrates the capability to change the pitch attitude of the aircraft while maintaining the flight path and velocity perturbations equal to zero. This maneuver can be useful in weapons release when a change in flight path is not desired or when coupled to a gun tracking controller for fine tuning of the tracking solution.

2. Vertical Translation. This maneuver attempts to control flight path while maintaining pitch angle and

velocity perturbations equal to zero. Vertical translation could be useful in a low airspeed, high AOA engagement where altitude can be gained through a reduction in AOA since the perturbation in theta is kept near zero (i.e $\gamma = -\alpha$).

3. Direct Climb. Conventional aircraft use the horizontal tail to rotate the fuselage thereby increasing AOA which results in an increase in flight path angle. A direct climb attempts to command theta and gamma equal to each other which forces the variation in alpha to zero throughout the maneuver. Again, the tactical application is to low airspeed, high AOA maneuvering where increases in alpha cannot be tolerated.

4. Constant g Pull-Up. This maneuver allows the pilot to command specific g loadings; a maneuver commonly used in air combat maneuvering. Since g capability is closely tied to pitch rate, thrust vectoring can play an important role in this maneuver. The constant g pull-up is not examined at 0.3 Mach/FL 200 because of the limited tactical utility at this flight condition.

3.7 Summary

The models presented in this chapter provide realistic control applications when demonstrating the capabilities of the Porter design method in the formulation of multivariable control laws. All of the assumptions

presented are consistent with good engineering practice and do not detract from the overall objectives of this design effort.

Chapter IV describes the details of the method used in the formulation of the longitudinal control laws based on the models presented in this chapter.

IV. Longitudinal Controller Design Method

4.1 Introduction

This chapter outlines the design procedures involved in the development of multivariable longitudinal control laws for the F-15/STOL derivative fighter. The details of the Porter design technique, implemented through the computer-aided design program MULTI, are presented as applied to the mathematical models developed in Chapter III.

The chapter begins with an examination of the model's controllability and observability, two fundamental requirements for a successful design. Following this, the importance of the output vector and its effect on the system's transmission zeros is described. Next, the complete design approach from the basic system to the full developed model (actuators, computational time delay, and sensors included) is presented. Finally, the effects of parameter variation and sensor noise on controller design and system stability is explained in detail.

Chapter V presents the maneuver simulation results using control laws designed by the procedures outlined in this chapter.

4.2 Controllability and Observability

The properties of controllability and observability must be present within the system representation in order to implement this design procedure. Controllability is a function of only the system's plant matrix \underline{A} and the input matrix \underline{B} . Specifically, for a system to be completely controllable, every state in the state vector \underline{x} at any time t_1 can be forced to its new value at time t_2 by an unconstrained input vector \underline{u} (7). For a linear, time-invariant system, this requires:

$$\text{Rank}[\underline{B} \quad \underline{A}\underline{B} \quad \dots \quad \underline{A}^{n-1}\underline{B}] = n \quad (4-1)$$

where n is the dimension of the system.

Observability requires that every mode of the system be present in every output. In other words, the state vector at time t_1 can be completely reconstructed from the measured output vector over the time interval $(t_2 - t_1)$. In state-space form, the plant matrix \underline{A} and the output matrix \underline{C} must be constructed so that:

$$\text{Rank}[\underline{C}^T \quad \underline{A}^T \underline{C}^T \quad \dots \quad \underline{A}^{T(n-1)} \underline{C}^T] = n \quad (4-2)$$

and again, n is the dimension of the system.

The multivariable control law theory that supports the Porter design technique offers an alternative approach for determining controllability and observability (20). As long as the invariant zeros of the system do not include

decoupling zeros (input, output, or input-output), then the system is guaranteed to be completely controllable and observable. The computer program ZERO quickly computes the invariant zeros of the state-space model.

All of the model configurations investigated in this thesis have no decoupling zeros and therefore, the open-loop models are both controllable and observable. Porter and Bradshaw have shown (18) that the addition of a proportional plus integral controller to a discrete-time, sampled data system does not change the closed-loop controllability or observability of the system. As a result, bounded inputs give bounded states which result in bounded outputs. This property is also referred to as global stability.

There is no requirement, however, for any system to be completely devoid of all invariant zeros. A certain subset of invariant zeros are transmission zeros which are present in all of the models examined in this study. This class of zeros is discussed in the next section.

4.3 Selecting the Output Vector

Determination of the output vector is a critical step in the design process for four reasons.

First, the output variables chosen for feedback must be physical variables readily available through aircraft sensor measurements. Since the Porter method uses

output feedback for control law formulation, the selected variables are all physical rates and angular accelerations commonly available to the pilot through cockpit instrumentation.

Secondly, selection of the output vector fundamentally influences the number and location of the system's invariant zeros. As mentioned previously, the lack of decoupling invariant zeros guarantees controllability and observability. In the three-state plant, the use of q and α feedback for the 1.4 Mach/FL 200 flight condition creates an output decoupling zero at $-.001$ and therefore is an unacceptable feedback vector. In addition, since $q = \dot{\theta}$, any output vector that includes q produces a transmission zero at the origin. The significance of this situation is discussed in the next section.

Third, all regular designs ($C_2 B_2$ has full rank) exhibit increased output decoupling as system gain is increased (Appendix B). For irregular designs which incorporate minor loop feedback through the measurement matrix M , output decoupling is dependent on the selected output vector. All of the designs in this thesis are irregular and therefore care is taken in the choice of the output variables.

As explained in Appendix B, the \tilde{T} transfer function matrix must be diagonal to achieve asymptotic output decoupling. The diagonal form is usually possible when the

variables associated with the kinematic equations are chosen for feedback. Theta is the kinematic state variable of the aircraft model used in this study. By selecting theta as an output variable, the $\tilde{\Gamma}$ matrix becomes:

$$\tilde{\Gamma} = \begin{bmatrix} \frac{3.33}{\lambda + 3.33} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (4-3)$$

when the measurement matrix is chosen as $\underline{M} = \{0.3, 0, 0\}^T$. A similar thesis by Acker (1) uses a feedback vector without the kinematic variable theta and the result is an off-diagonal term in the $\tilde{\Gamma}$ matrix. Consequently, full output decoupling can never be achieved, no matter how high the gain.

It is important to realize that this limitation may have no practical consequence since infinite gain systems are never implemented. As a result, the responses using the output scheme employed by Acker may be no different than the results obtained from a system capable of pure asymptotic decoupling.

Finally, certain combinations of output variables are not permitted by the design method for this particular model. Because of the dependence between q and $\dot{\theta}$ ($q = \dot{\theta}$), these two variables cannot be included in the same

output vector. Mathematically, this is demonstrated as follows:

$$\underline{F}_2 = \underline{C}_2 + \underline{MA}_{12} \quad (4-4)$$

and \underline{F}_2 must have full rank since $(\underline{F}_2 \underline{B}_2)^{-1}$ must exist for irregular designs. For the three-state plant with $\underline{y} = (\theta, q)^T$, Equation (4-4) becomes

$$\underline{F} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (4-5)$$

Obviously, \underline{F}_2 can never have full rank for any values of m_1 or m_2 . In the four-state plant with $\underline{y} = (\theta, q, \alpha)^T$, a similar situation exists:

$$\underline{F}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \quad (4-6)$$

where \underline{F}_2 cannot have full rank for any vector \underline{m} . As a result, velocity, u , must be a part of any output vector selected. This is exactly the reason why the MCAIR four-state model is augmented with a thrust input. Since velocity must be fed back, there must be an input device that provides control.

Transmission Zeros. In theory, the transmission zeros of a linear, multivariable system define regions in the complex s-plane where the finite "slow" roots (poles) of the characteristic equation will migrate under conditions of high gain. In the discrete case, high gain is equal to the high sampling frequency ($f = 1/T$). Consequently, under infinite gain conditions and with stable transmission zeros, system stability is guaranteed to be achievable for all bounded inputs. Most systems, however, are stable at finite gain values.

It has been shown that for systems that have an equal number of inputs and outputs (the only type that the program MULTI allows), the number of transmission zeros is equal to:

$$\# \text{ of } Z_t = (n-m) - \text{Rank Deficiency of } (\underline{C}_2 \underline{B}_2) \quad (4-7)$$

where n = the number of states and m = the number of inputs (20). As explained earlier (Chapter III), the four-state plant for the F-15/STOL can have a maximum of three inputs and therefore only three outputs.

It would seem from Equation (4-7) that with $n = 4$, $m = 3$, and a rank deficiency of one in $\underline{C}_2 \underline{B}_2$, there would be no transmission zeros. This is not the case, however, since a rank deficiency in $\underline{C}_2 \underline{B}_2$ requires the addition of a measurement matrix \underline{M} (irregular design) whose reciprocal elements define additional transmission zeros in the system.

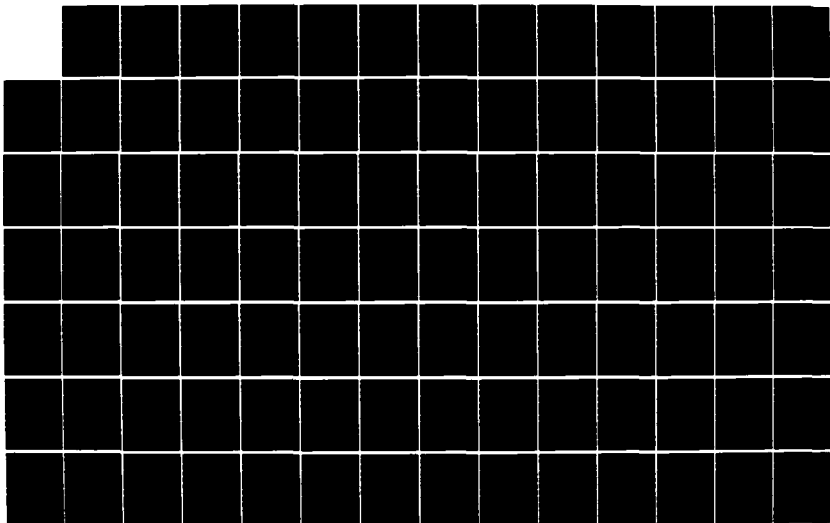
AD-A164 017

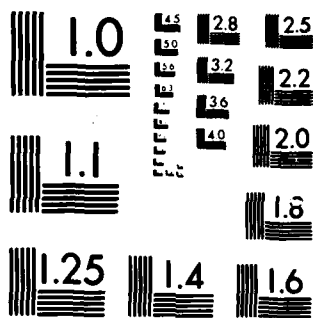
MULTIVARIABLE CONTROL LAW DESIGN FOR ENHANCED AIR
COMBAT MANEUVERING F-15. (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. K A SHEEHAN
DEC 85 AFIT/GE/EE/85D-38 F/G 1/2

2/5

UNCLASSIFIED

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

(Appendix B). The advantage to this approach is that the location of the zeros are chosen by the designer given the elements selected in the measurement matrix. This flexibility is useful in one of the designs discussed later in this chapter. By selecting q as a feedback variable, however, a "regular" design results which has no rank deficiency and consequently no requirement for a measurement matrix. The drawback is that the single transmission zero is located at the origin with no capability to reposition it.

The significance of a transmission zero at the origin is twofold. First, it indicates that for all bounded inputs, one state is unbounded. In the practical sense, if q is given a step input, then θ ramps to infinity but at a very predictable rate ($\dot{q} = \dot{\theta}$). This is obviously a very desirable outcome and in practice, q is only given a pulse by the pilot until a desired pitch angle is reached. A second result of a transmission zero at the origin is the potential trouble it may cause in the time response of certain designs. The ability to move the zero to achieve a satisfactory time response is critical in the pitch pointing controller at 0.3 Mach/FL 200. An example of the zero's effect is given in Section 4.4, Tailoring of Input Responses. This study avoids pitch rate feedback in the output vector because of the design inflexibility resulting from the transmission zero at the origin.

4.4 Design Approach

A systematic procedure is followed in this study for the development of longitudinal controllers for the F-15/STOL model. This design procedure is refined by many trial-and-error design attempts that failed to produce acceptable results. It should be noted, however, that this approach is successful with the model used in this study but is not necessarily applicable to models exhibiting different transfer function characteristics.

Maximum Maneuver Capability. The first step in the design process is the tailoring of an input vector appropriate for the desired maneuver. Because of the control input limits, however, each maneuver has a maximum capability. In order to determine this maximum prior to attempting a simulation, the steady-state transfer function matrix is calculated.

For any linear, time-invariant system, the output vector is related to the input vector by a transfer function that, in the Laplace domain, is given by

$$\underline{y}(s) = \underline{G}(s) \underline{u}(s) \quad (4-8)$$

In the case of integral control with output feedback (Porter method), the output vector \underline{y} is identically equal to the input vector \underline{u} in steady-state when the system is driven by a step input. Applying the final value theorem

to Equation (4-8), the steady-state value for the output vector is given as:

$$\underline{y}(t)_{ss} = \underline{G}(o) \underline{u}(t) \quad (4-8a)$$

Using the condition defined above for zero steady-state error given a step input, \underline{y} is replaced by \underline{v} and Equation (4-8a) is now written as:

$$\underline{u}(t)_{ss} = \underline{G}(o)^{-1} \underline{v}(t) \quad (4-8b)$$

Equation (4-8b) gives the relationship between the control input \underline{u} and the pilot input \underline{v} under steady-state conditions. This calculation is incorporated into the program MULTI as a user option. Table 4.1 lists the maximum maneuver capability at each flight condition for each of the three four-state model maneuvers. The constant g pull-up uses the three-state model plant and is not included, since maximum maneuvering exceeds the linear assumptions defined in Chapter III.

In addition to preventing the control inputs from exceeding their deflection limits, control input rates must also be observed. The high gain characteristics of the Porter method lends itself to rapidly responding control inputs. The input rates can be controlled by ramping the pilot input vector \underline{v} to its steady-state value. All of the \underline{v} inputs are initially ramped to steady-state in 0.8 secs.

TABLE 4.1
MAXIMUM MANEUVER CAPABILITY

Flight Condition	Pitch Pointing	Vertical Translation	Direct Climb
0.3M/FL200	2.8 degs (-)Canard*	2.2 degs (+)Stabilator*	1.8 degs (+)Canard*
0.9/FL200	2.9 degs (-)Canard*	1.9 degs (+)Canard*	2.0 degs (+)Canard*
1.4M/FL200	2.1 degs (-)Canard*	1.1 degs (+)Canard*	2.0 degs (+)Canard*
2.0M/FL400	1.9 degs (-)Canard*	0.98 degs (+)Canard*	2.0 degs (+)Canard*

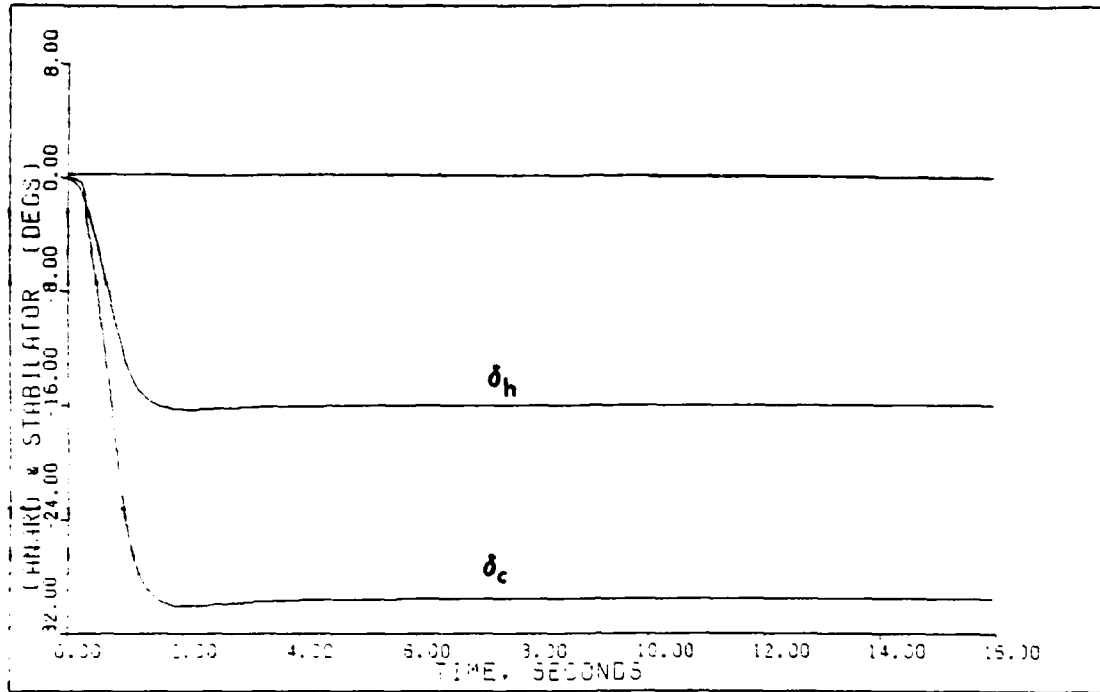
*limiting control surface.

The results of this first approximation proved satisfactory for all but one maneuver at one particular flight condition (Chapter V).

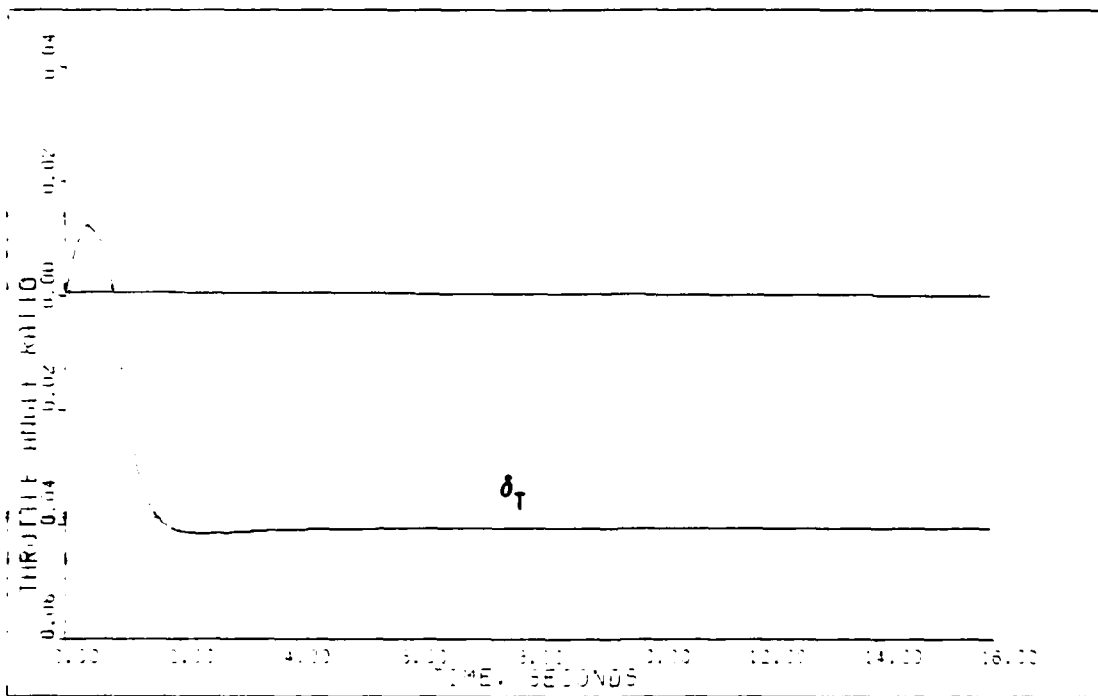
The "sign-swapping" problem affecting the \dot{u} equation control derivatives for the canard and stabilator (see Chapter III) is evident using the $\underline{G}(0)$ option in MULTI. For the 0.9 Mach/FL 200 flight condition, commanding a +2.9 degs in theta for the pitch pointing maneuver gives the following steady-state control inputs: canard = -29.56 degs, stabilator = -15.99 degs, and throttle = -.0407. This demonstrates that for a maneuver that increases overall drag, a reduced steady-state drag is specified, which is not realistic. Upon close inspection, the error is found

in the stabilator. The sign of the stabilator's control derivative is negative which, when multiplied by a negative control deflection, produces the equivalent of thrust in the positive x direction which is impossible. As a result, the throttle is reduced from its equilibrium value to balance the forces. Figures 4.1a and 4.1b show the results of a linear simulation if this condition is present. The predicted results from the $\underline{G}(0)$ inverse calculations are supported by the simulation.

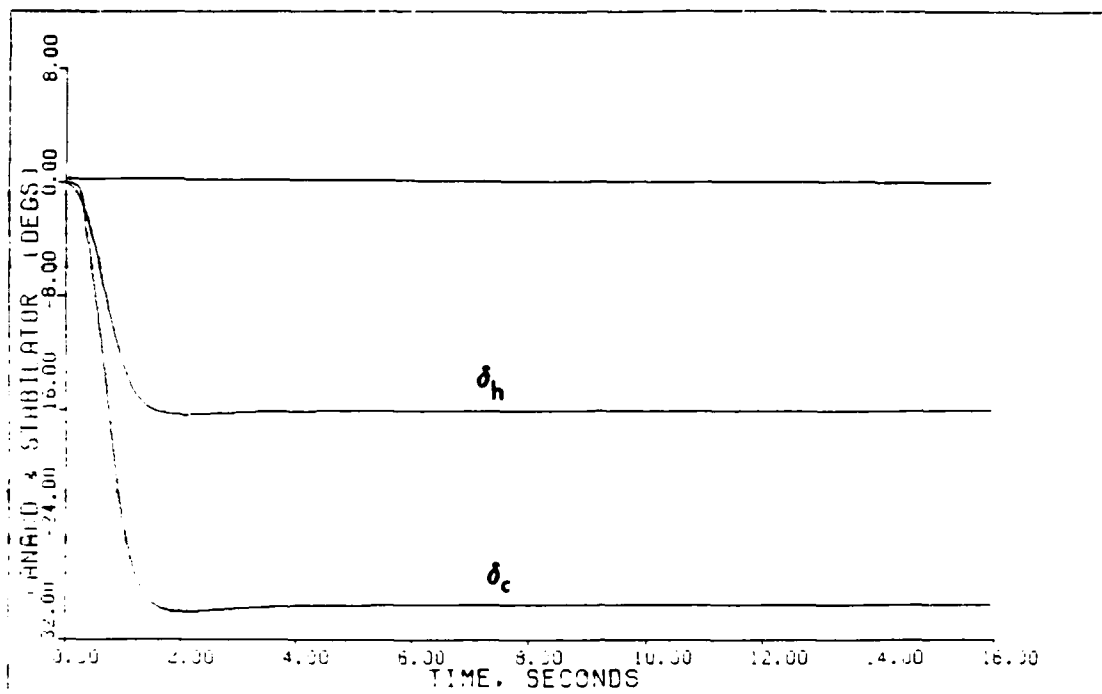
The "fix" to this problem is to perform a nonlinear simulation by testing the zero angle of attack condition of each aerodynamic surface at each sample time. As the surface transits this condition (i.e. the deflection goes negative with a negative control derivative), the sign of the derivative is reversed to prevent the surface from "creating" thrust. By changing the sign of the stabilator derivative in the \underline{B} matrix and recalculating the $\underline{G}(0)$ inverse matrix, MULTI gives the following steady-state control deflections: canard = -29.56 degs, stabilator = -15.99 degs, and throttle = +.0831. The throttle now correctly increases to account for the increase in drag--a more satisfying result. Figures 4.2a and 4.2b illustrate the effects of the nonlinear simulation on the control inputs. The results show that the canard and stabilator still deflect to the same values as before; however, now the throttle has increased to provide the required thrust.



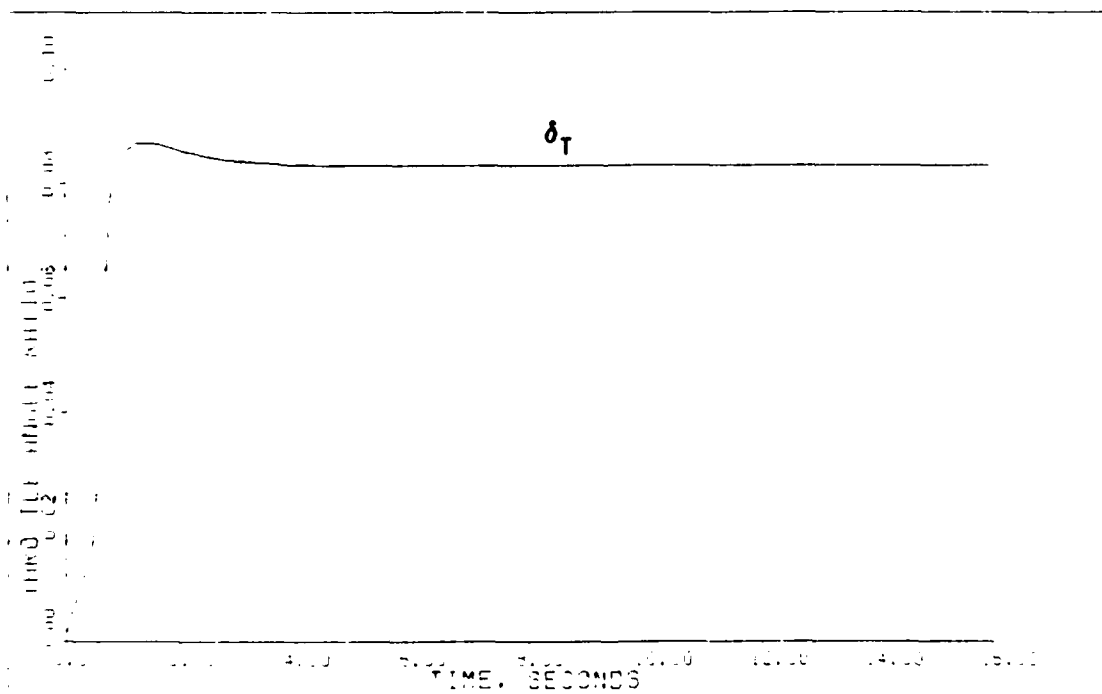
PITCH-POINTING: LINEAR SIMULATION 0.9M. FL200 Fig. 4.1a



PITCH-POINTING: LINEAR SIMULATION 0.9M. FL200 Fig. 4.1b



PITCH POINTING: NONLINEAR SIMULATION (0.9M, F1000) Fig. 4.2a



PITCH POINTING: NONLINEAR SIMULATION (0.9M, F1000) Fig. 4.2b

Achieving Stability. The Porter method uses the MULTI design program to create the gain matrices for the proportional plus integral controller, \underline{K}_0 and \underline{K}_1 respectively. The relationship between \underline{K}_0 and \underline{K}_1 is defined as follows:

$$\underline{K}_0 = \epsilon (\underline{C}_2 \underline{B}_2)^{-1} \underline{\Sigma} \quad (4-9a)$$

$$\underline{K}_1 = \bar{\alpha} \underline{K}_0 \quad (4-9b)$$

where $\bar{\alpha}$ and ϵ are gain multipliers with $\underline{\Sigma}$ as the input-output, diagonal weighting matrix which must be selected by the designer. In most cases stability is achieved by adjusting ϵ to a satisfactory value with all other design parameters set to unity. This area of conditional stability is difficult to find in some designs exhibiting open-loop static instability (6). A further adjustment of the sigma weighting matrix may be required to achieve initial stability. The design parameter ϵ is chosen merely as a design convenience since it proportionally affects both \underline{K}_0 and \underline{K}_1 . The same effect is realized by proportionally scaling the diagonal elements of the sigma matrix.

Tailoring the Input Responses. After stability is achieved, the next task in the design process is to check the time responses of the \underline{u} vector, the control inputs. Frequently, with unity elements in the sigma matrix, the control inputs respond so quickly that both rate and

deflection limits are exceeded. At this point, a fine tuning of the controller must take place, with several options available to the designer.

As already mentioned, the elements of the diagonal sigma matrix play an important part in both the system stability and the transient response of the output variables. Under conditions of high gain, the "slow" modes of the system with finite roots become uncontrollable and unobservable as they approach the regions of the transmission zeros. The "fast" modes with infinite roots become dominant in the transient response and for regular designs, exhibit increasingly non-interactive, or decoupled behavior as the gain is increased. Certain irregular designs also exhibit decoupled characteristics as explained earlier in this chapter.

Under conditions of decoupling, the value of the Porter method becomes apparent. Each diagonal element of the weighting matrix affects the transient response of its corresponding output variable ($\sigma_1: y_1$, $\sigma_2: y_2$, etc.) with minimal interactive effect on the remaining outputs. The initial problem in the design, however, is how to get the control input responses within their respective rate and deflection limits. The control input rate problem is generally solved by ramping the system inputs to their steady-state values and "smoothing" the corners with an option available in MULTI.

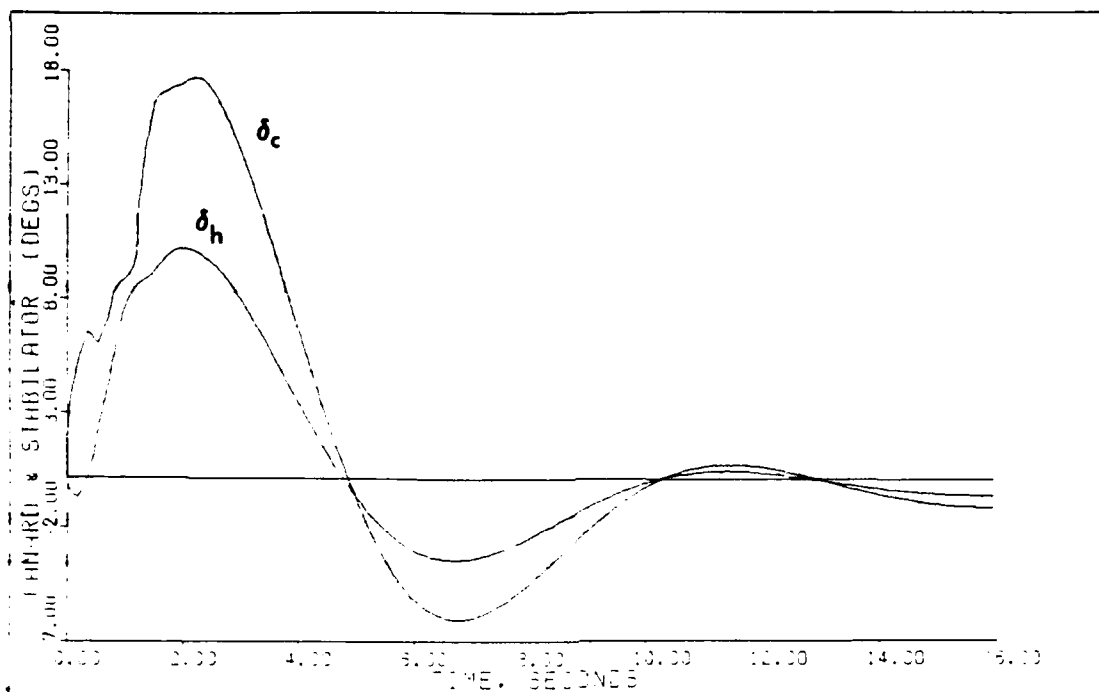
The \underline{u} input overshoot problem, however, does not have a straightforward solution since there is no correlation between the design parameters and the control input responses. A previous thesis, using the X-29 model, concluded that the relative magnitudes of the columns of the $\underline{G}(0)$ inverse matrix provide insight into picking values for the weighting matrix (6). There is no mathematical reason to support this conclusion since all of the information about the transient characteristics of the response is lost when the final value theorem is applied to form this steady-state matrix. If the problem can be solved by the sigma matrix, it is accomplished by a systematic method of trial and error in adjusting the relative magnitudes of the diagonal elements. The price that is paid for a relatively low sigma value (as compared to the other diagonal elements) is found in the output response which is covered in the next section.

Another technique for tailoring \underline{u} vector responses is to reduce the amount of integral gain used in the controller. By reducing the integral-to-proportional gain constant, $\bar{\alpha}$, the value of \underline{K}_1 matrix is reduced (see Equation (4-9b)) and, consequently, the system no longer tries to respond as quickly to each commanded input.

This is a very effective technique in reducing the \underline{u} vector overshoot, but the drawback is that the system takes longer to reach steady-state (zero error between

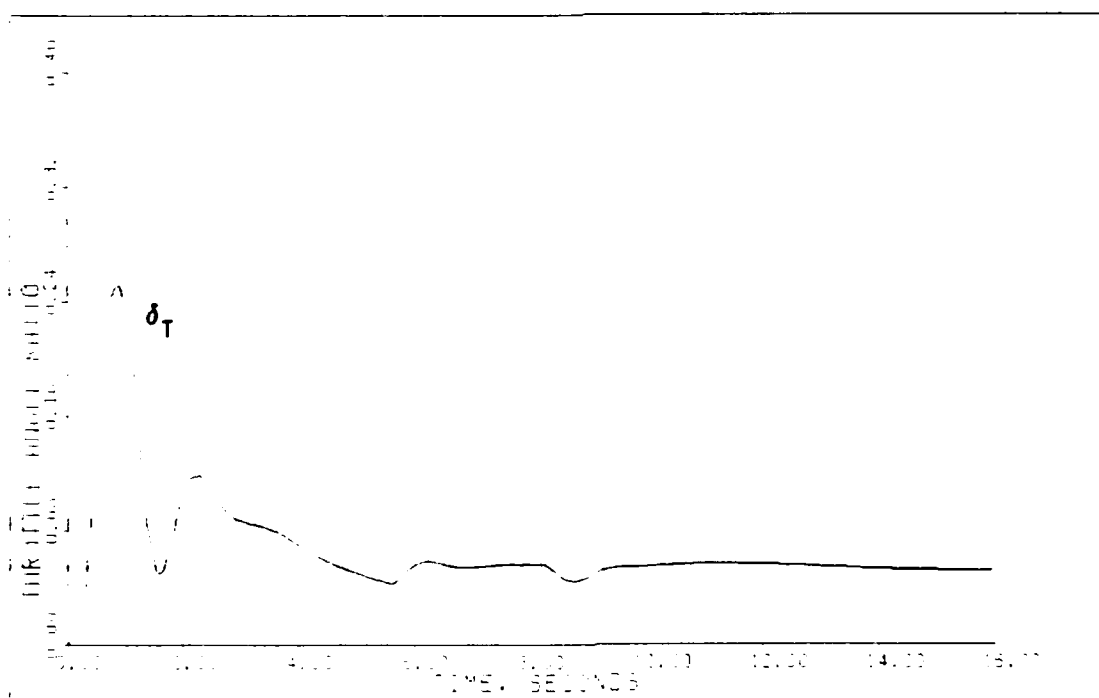
output and input). For an aircraft controller, this is a more acceptable alternative than a highly oscillatory response that has a shorter settling time. In the latter case, a pilot may overcontrol the aircraft while trying to damp the oscillations in the controller response. Figures 4.3a and 4.3b show the \underline{u} vector responses with relatively high integral gain. In contrast, Figures 4.4a and 4.4b are the plots of the same controller at the same flight condition with a much lower integral gain. The responses in the latter case are much smoother and more well behaved. It should be noted that this technique is not well suited to either the pitch pointing or vertical translation maneuvers since errors in the output variables that are commanded to zero do not die out quickly and therefore reduce the maneuver's effectiveness.

Another option available to the designer for controlling the input responses is the use of the measurement matrix in "irregular" designs. As described earlier, in the case of a minimally populated measurement matrix (fewest nonzero elements), the inverse of each element determines the location of a transmission zero of the system. As a result, the position of the transmission zeros can be altered by the selection of the measurement matrix elements. In certain designs, this can have a profound effect on the transient response of the system. By moving the zero closer to the origin, the response time is



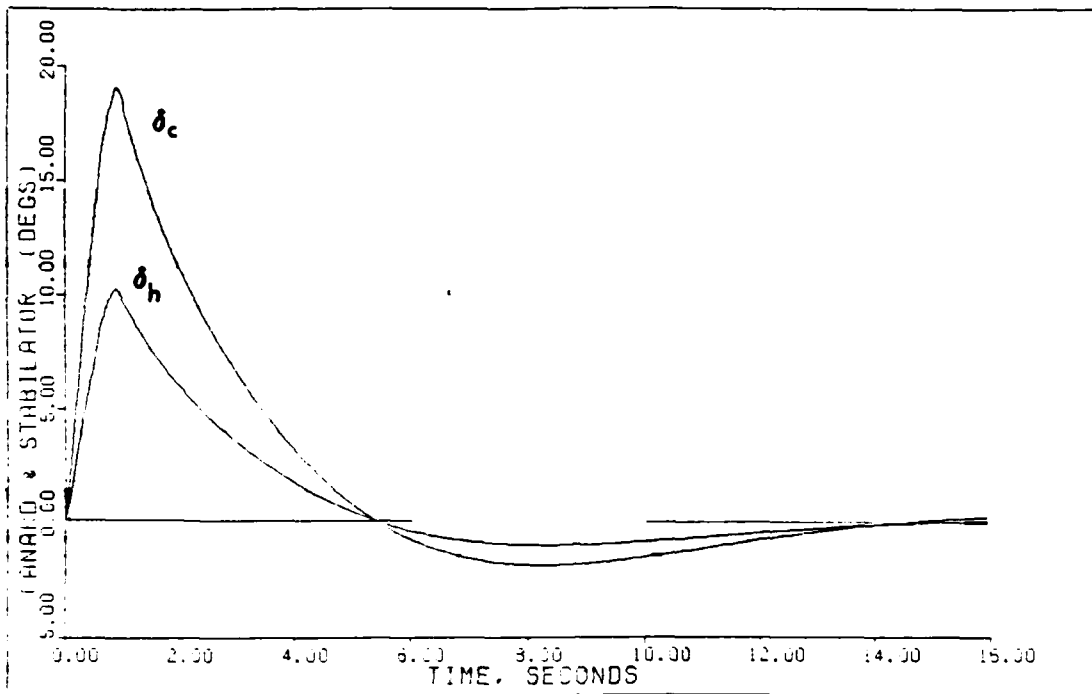
DIRECT CLIMB: HIGH INTEGRAL GAIN. $K_{I\delta} = 6.67 \times 10^{-3}$ M. FL100

Fig. 4.3a



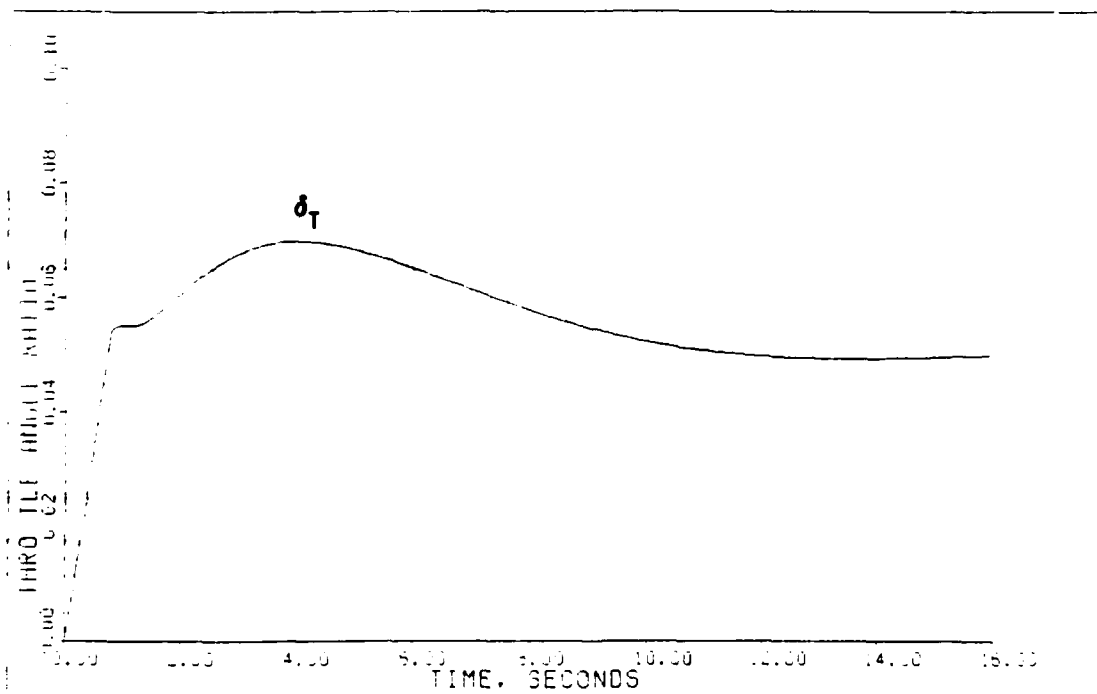
DIRECT CLIMB: HIGH INTEGRAL GAIN. $K_{I\delta} = 6.67 \times 10^{-3}$ M. FL100

Fig. 4.3b



DIRECT CLIMB: LOW INTEGRAL GAIN, $K_I = 0.001 \cdot K_D$ (0.3M, FL000)

Fig. 4.4a



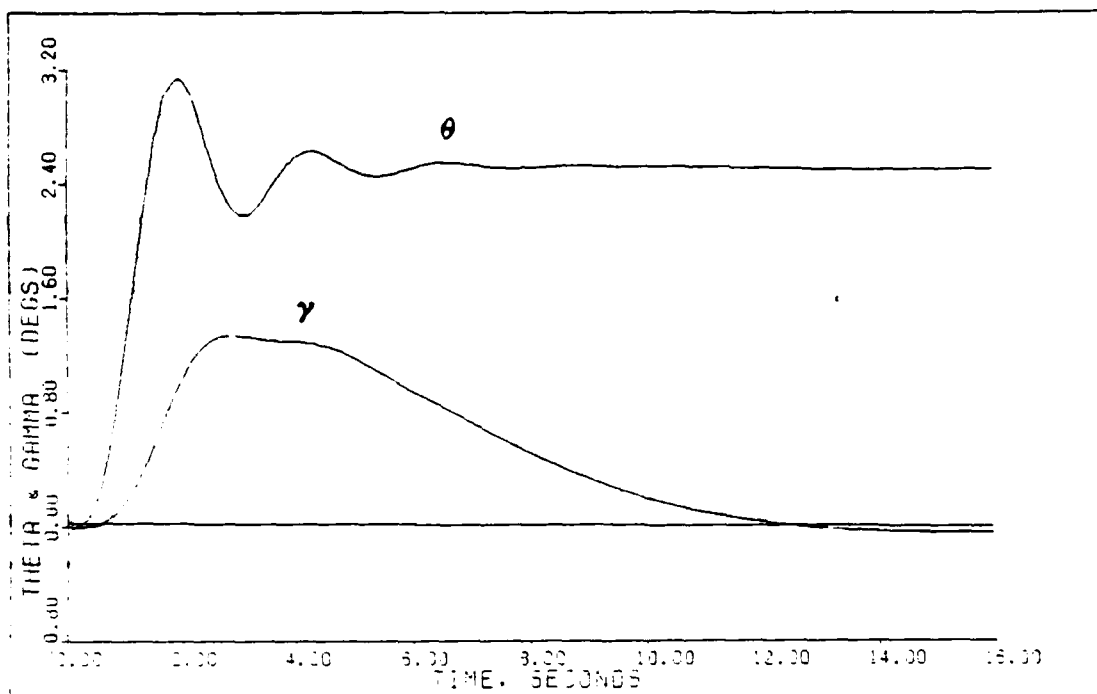
DIRECT CLIMB: LOW INTEGRAL GAIN, $K_I = 0.001 \cdot K_D$ (0.3M, FL000)

Fig. 4.4b

increased (slower response) which reduces control input overshoot. The complication that results from this technique is one of implementation. Gain scheduling would now be necessary in the \underline{M} matrix as a function of flight condition.

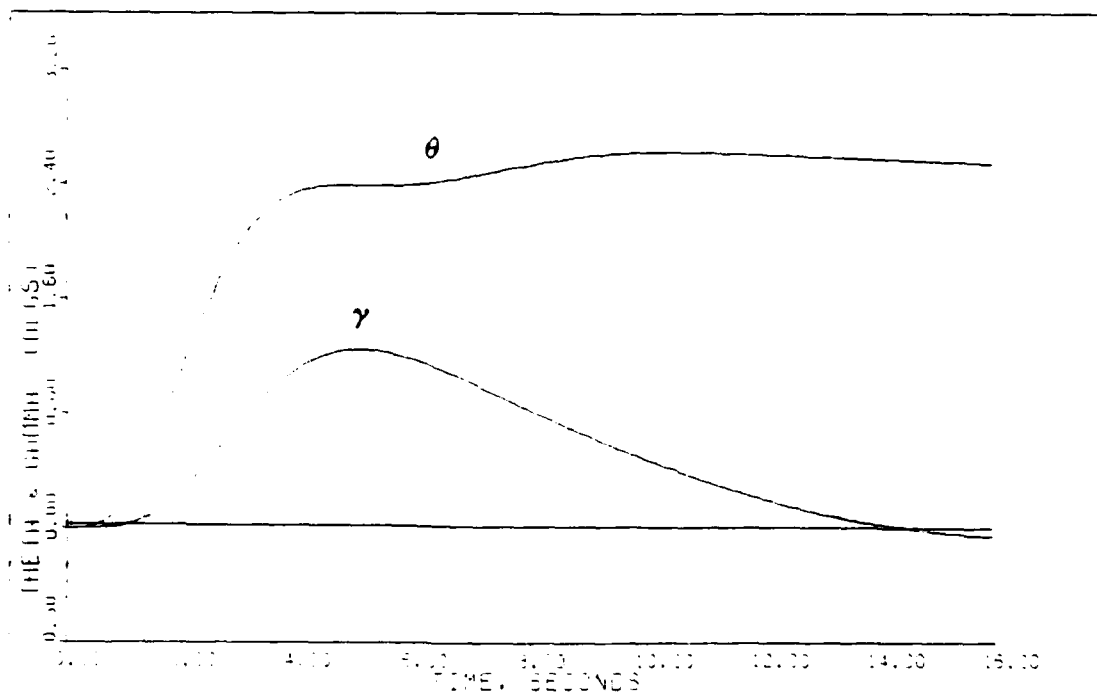
Tailoring of the Output Responses. After the control inputs are tailored to lie within acceptable rate and range limits, the control laws are further refined to give acceptable output responses. The technique of using the transmission zero to control system response can be applied here as well. Figure 4.5a shows the control input responses with a single transmission zero located at -3.33 ($\underline{M} = (0.3, 0, 0)^T$). The output theta shows a second order, underdamped response which could be very easily overcontrolled by the pilot. By shifting the zero closer to the origin, the response is damped with approximately the same settling time as the previous example (Figure 4.5b). From the pilot's standpoint, this is a more desirable response.

The common method of adjusting the transient response of the output variables is done with the sigma matrix. As described previously, under decoupled conditions (high gain) the diagonal elements of the weighting matrix uniquely control the transient characteristics of their corresponding outputs. As a guide, the higher the value of the sigma element, the faster the response of the



PATCH POINTING: TRANSMISSION ZERO = -6.33, 0.3M, F1000

Fig. 4.5a



PATCH POINTING: TRANSMISSION ZERO = -6.476, 0.3M, F1000

Fig. 4.5b

output. Again, the limitation on the speed of response is generally dictated by the rate and deflection limits of the input controls.

The final adjustment to the controller design is made by setting the integral-to-proportional gain factor ($\bar{\alpha}$), to an acceptable level that gives reasonable settling times for the output variables. From Equations (4.9a) and (4.9b), the proportional gain is unaffected by any changes to $\bar{\alpha}$. Older versions of MULTI included $\bar{\alpha}$ in the computation of K_0 . As a result, both ϵ and $\bar{\alpha}$ had to be readjusted each time the integral gain was changed. These equations are now modified in the current version of MULTI so as to correspond to Equations (4.9a) and (4.9b).

For most designs the value of $\bar{\alpha}$ is increased to a point short of inducing overshoot in the output or exceeding the limits of the input controls.

Model Development. The design of a controller for the basic aircraft is generally a straightforward task when using the procedure described in this chapter. The real system, however, incorporates additional complexities such as actuator and sensor dynamics, computational time delay, and noise corruption. System noise is generally present in both the outputs and the system states. Output noise results from noise corrupted sensors while the noise in the aircraft states is generally caused by random wind gusts and wind shears.

In an effort to accurately represent the real world, these complexities are included in the design process. Each of the delays is added in a "building block" approach, with the controller being re-tuned at every step. This process gives good results and leads to an acceptable design in a minimum amount of time.

The problem with this part of the design process is the lack of insight on the part of the designer as each delay is added to the overall model. The current version of MULTI calculates the closed-loop roots of the basic plant and any additional roots resulting from the integral controller. MULTI never computes the new closed-loop roots added to the overall transfer function resulting from the actuator and sensor dynamics. Consequently, very slow instabilities are extremely difficult to detect in the output responses but would be readily apparent from the closed-loop roots. This improvement is included as a recommendation for future work.

4.5 Parameter Variation

As described in Chapter III, the aircraft model is a linear approximation that is valid only for small perturbation analysis about an equilibrium point within the flight envelope. Modern day fighter aircraft operate in a sizeable flight envelope that encompasses altitudes from sea level up to FL 500 and speeds that range from 100 knots

to twice the speed of sound. To accomplish a valid design, numerous equilibrium points must be used to adequately cover such an expansive flight envelope.

Obviously, only a finite number of points can be used in any realistic design. As a result, the controller's characteristics are unknown in regions of the envelope in between the design points. Therefore, an important property of any design is insensitivity, of some degree, to changing plant parameters resulting from excursions away from the design flight condition.

This insensitivity to parameter variation, sometimes referred to as "robustness," can be demonstrated in a number of ways. One technique is to use a controller from one flight condition in simulations involving model data from different design points. The outputs from these simulations demonstrate a measure of controller robustness. This is not a realistic approach in this study since the design points are so widely separated within the flight envelope. Plant coefficients change in excess of an order of magnitude between certain data points which places unrealistic constraints on any design method that attempts to accommodate these variations.

An alternative method used in this study quantitatively measures the sensitivity of the controller as a function of the percentage change of a single control derivative within the plant. No physical significance is

attached to the variation of the selected derivatives. Most physical effects, such as control input failure sustained through battle damage, complicates the model by a simultaneous change in numerous derivatives while introducing lateral-longitudinal cross-coupling effects. This situation is beyond the scope of this thesis. Chapter V includes the results of the sensitivity analysis using the vertical translation maneuver at 0.9 Mach and FL 200.

In addition to robust characteristics, it is highly desirable to achieve a single controller design at one flight condition that can perform any of the desired maneuvers with acceptable results. This eliminates the requirement that the aircraft know a priori what maneuver the pilot is about to perform. Chapter V presents the results of the direct climb controller used in simulations performing both the pitch pointing and vertical translation maneuvers. The capability of a single controller that performs all three maneuvers is demonstrated at two design points (1.4 M/FL 200 and 2.0 M/FL 400). Time constraints prevented the completion of the analysis at all four flight conditions. Regardless, the results at only two flight conditions adequately demonstrates the flexibility of the controller design.

Finally, since the system's sampling rate is much faster than the rate of change in the stability derivatives, gain scheduling is an acceptable method to handle parameter

variation. Gain scheduling becomes particularly attractive when considering the alternative which accepts the inevitability of reduced performance in exchange for a control design that maintains stability despite large parameter variations.

4.6 Noise Effects

Aircraft sensor noise is frequently modeled as independent, zero mean, white gaussian noise that is injected at the outputs prior to feedback (14). As a result of this thesis and a similar study, modifications made to MULTI allow for the incorporation of sensor or disturbance noise into the simulation (1). Typical sensor noise values were obtained from a previous thesis using the LQG design method with the Navy F-14 Tomcat as a model (17). Appendix A describes the details involved in this change to MULTI.

From the design standpoint, controller parameters are not changed after the final iteration which accommodates sensor dynamics into the model. With respect to noise effects, the goal of this study is to examine the effects of sensor noise on system stability and performance. The pitch pointing maneuver at 1.4 Mach/FL 200 is used to study these effects (Chapter V). The typical noise values for θ , u , and γ are used as a starting point and then proportionally increased until control surface divergence occurs.

This gradual increase in noise level determines the maximum system noise tolerance.

This simulation method uses unfiltered, white gaussian noise in the feedback channel which is then passed to the proportional plus integral controller. The integration of white gaussian noise results in Brownian motion and is observed in the control surface as "random walk" at higher levels of noise (Chapter V). This characteristic is expected but could be reduced by the addition of a noise filter prior to the integration. The filter would present time-correlated noise to the controller due to the limited bandwidth of the filtered noise.

This improvement was not necessary for the purposes of this study but is recommended for future work.

4.7 Summary

This chapter outlines the methodology used in the design of longitudinal control laws for the F-15/STOL aircraft. Two basic models, as described in Chapter III, are used to perform four simulation maneuvers. The three-state model is used only for the constant g pull-up maneuver. The design methods described in this chapter apply equally well to both models.

As mentioned earlier, this chapter does not define a specific design sequence that would have universal application to all future Porter designs. The reason being

that there is no single method for all designs since the characteristics of each model are unique. What is presented is a suggested method of attacking the design problem, supplemented with numerous techniques for achieving satisfactory results. In summary, these general design steps are:

1. Achieve stability
 - usually accomplished through an overall adjustment gain.
2. Tailor input responses
 - various techniques are offered that influence both rate and deflection limits.
3. Tailor output responses
 - the characteristics of the transient response are fine tuned.
4. Adjustments, such as additional dynamics, are added
 - the design parameters are modified as more complexity is added to the model in a "building block" approach.

The next chapter details the results achieved by applying the techniques developed in this chapter.

V. Longitudinal Control Law Design Results

5.1 Introduction

This chapter presents the simulation results of the longitudinal control laws developed for the F-15/STOL aircraft using the methodology described in the previous chapter. The three maneuvers using the four-state model are: direct climb (0.3 Mach/FL 200), vertical translation (0.9 Mach/FL 200), and pitch pointing (1.4 Mach/FL 200). The single maneuver using the three state model is the constant g pull-up (2.0 Mach/FL 400). Each of the four flight conditions is represented in this chapter using a single maneuver for illustrative purposes. The remaining results for each maneuver are included in Appendix D.

The parameter variation results are presented in Section 5.6 in two parts. First, the direct climb controller is used to perform both the vertical translation and pitch pointing maneuvers at a single flight condition. Although plant parameters do not vary in this demonstration, controller capability is displayed by handling a variety of command inputs. These results are given at both 1.4 Mach/FL 200 and 2.0 Mach/FL 400.

Secondly, the vertical translation maneuver is selected at 0.9 Mach/FL 200 to demonstrate controller robustness to parameter variation. The three control

derivatives, M_{δ_C} , M_{δ_H} , and X_{δ_H} , are varied independently while plotting their effects on the system response. The results of this analysis demonstrate a qualitative measure of robustness in the control law design.

Finally, the effects of sensor noise on system performance is shown using the pitch pointing maneuver at 1.4 Mach/FL 200. The results include response plots at both typical noise values and higher values approaching the maximum capabilities of the system.

5.2 Direct Climb (0.3 Mach/FL 200)

The direct climb maneuver is accomplished by commanding both the pitch and flight path angles to the same values, which forces the perturbation in alpha to zero. Unlike the other maneuvers, the direct climb is limited not by the steady-state control surface deflections but by their peak transient deflections. Since the change in velocity equals zero in the steady-state, the control surfaces return to their equilibrium values with the aircraft established in a steady climb at the commanded flight path angle. A 2.0 deg direct climb is commanded for this flight condition.

Table 5.1 lists the design parameters at each of the four stages leading to the final design. This data provides insight into the evolution of the design as more complexity is added to the basic model. Table 5.2

summarizes the pertinent figures of merit of the output responses. Figures 5.1 through 5.20 depict the system's time response during this maneuver.

As mentioned earlier, the peak transient response of the aerodynamic surfaces determines the maximum magnitude of this maneuver. Figures 5.1 through 5.4 show that the canard is the limiting control in each stage of the design. The canard deflects in the positive direction to produce the necessary aircraft rotation while the positive deflection of the stabilator helps control the moment and reduce the transient in α . The inputs movements are smooth and quite fast, approaching the rate limit of the canard. As expected, the final value of the control surfaces returns to zero (equilibrium value) as the new flight path is reached. The throttle smoothly increases to provide the additional thrust required for the climb.

Figures 5.9 through 5.20 indicate the relatively slow output response to the rapid control inputs. This behavior is predictable for two reasons. First, the low dynamic pressure at this flight condition makes the controls less effective than at higher Mach numbers. Secondly, very low integral gain ($\underline{K}_1 = (.01)\underline{K}_0$) is used at this design point to reduce the canard's peak transient. A higher integral gain would demand a faster rise time and shorter output settling time which could only be achieved by excessive deflection of the control inputs.

Figures 5.9 and 5.13 seem to indicate that there is a steady-state error in the velocity and pitch angle responses. This is not the case since the presence of integral control forces zero steady-state error to a step input in a finite time interval. Because of the relatively low integral gain, the settling time is greater than the simulation time. In addition, since the sigma value for theta (σ_1) is much smaller than the sigma value for gamma (σ_3), the flight path reaches its commanded value much sooner than the pitch attitude (Figures 5.13 and 5.17).

One of the aerodynamically pleasing features of this particular maneuver is the response of angle of attack (Figure 5.9). The capability to decouple the outputs allows for a very small transient in alpha which is critical at low speed, high AOA conditions. At trimmed conditions approaching the stall AOA, this maneuver would not be possible with a conventionally configured aircraft.

Figures 5.2, 5.6, 5.10, 5.14, and 5.18 show the effects of actuator dynamics on the system response. Since their dynamics are well outside the bandwidth of the basic plant, their effects are negligible on overall system performance.

The next step in the "building block" method of controller design is the addition of computational time delay. This simulates a worst case condition since the equivalent of one sample period delay is now present in

TABLE 5.1
DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Direct Climb (+2.0 degs)

Flt Condition: 0.3 Mach at FL 200

Command Vector y : v_1 = Theta: 0.8, 0.03491, 20, 20
 v_2 = Velocity: 0, 0, 0, 0
 v_3 = Gamma: 0.8, 0.03491, 20, 20

Basic Plant

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>K_0</u>		
0.010	0.010	0.5	.5341E+00	.0000E+00	.1727E+02
		2.5	-.1505E+00	.0000E+00	.9788E+01
		1.64	.3550E-03	.1250E-02	.4561E-01

Plant + Actuators

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>K_0</u>		
0.010	0.010	0.5	.5341E+00	.0000E+00	.1727E+02
		2.5	-.1505E+00	.0000E+00	.9788E+01
		1.64	.3550E-03	.1250E-02	.4561E-01

Plant + Actuators + Delay

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>K_0</u>		
0.010	0.020	0.5	.1068E+01	.0000E+00	.3454E+02
		0.9	-.3011E+00	.0000E+00	.1958E+02
		1.64	.7101E-03	.9000E-03	.9121E-01

Plant + Actuators + Delay + Sensors

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>K_0</u>		
0.010	0.020	0.5	.1068E+01	.0000E+00	.3454E+02
		0.9	-.3011E+00	.0000E+00	.1958E+02
		1.64	.7101E-03	.9000E-03	.9121E-01

Notes:

1. Each v input is composed of four parts:
 - A. Time (secs) that the input reaches steady state.
 - B. Steady-state value (radians).
 - C. Time (secs) input leaves steady-state.
 - D. Time (secs) input reaches zero.
2. Sigma = the elements (in order) of the diagonal matrix.
3. The integral controller matrix $K_1 = (\alpha)K_0$.
4. Irregular Design: $\underline{M} = (0.3, 0, 0)^T$.

TABLE 5.2
DESIGN OUTPUT FIGURES OF MERIT

Maneuver: Direct Climb (+2.0 degs)

Flt Condition: 0.3 Mach at FL 200

<u>Basic Plant</u>			
<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	+2.301	7.175	14.00
Velocity	-1.294	5.425	**
Flight Path Angle	+2.174	8.225	13.47

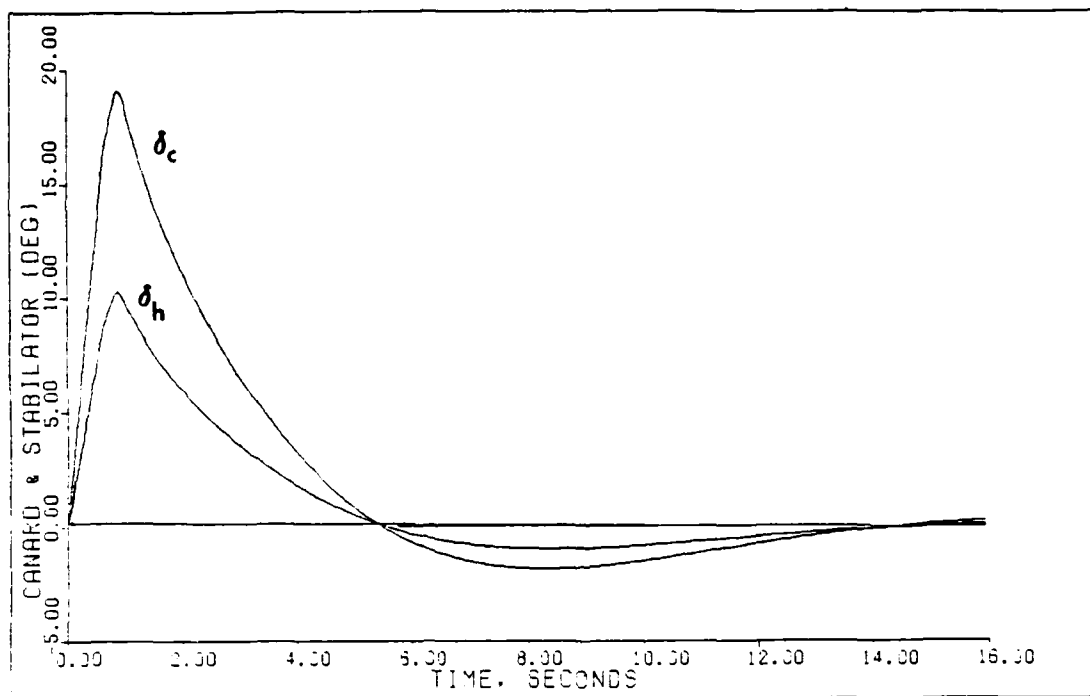
<u>Plant + Actuators</u>			
<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	+2.316	7.175	14.00
Velocity	-1.308	5.425	**
Flight Path Angle	+2.180	8.225	13.47

<u>Plant + Actuators + Delay</u>			
<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	+2.355	7.350	14.17
Velocity	-3.098	7.525	**
Flight Path Angle	+2.197	8.225	13.47

<u>Plant + Actuators + Delay + Sensors</u>			
<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	+2.456	6.825	13.47
Velocity	-1.425	5.250	**
Flight Path Angle	+2.248	7.700	13.47

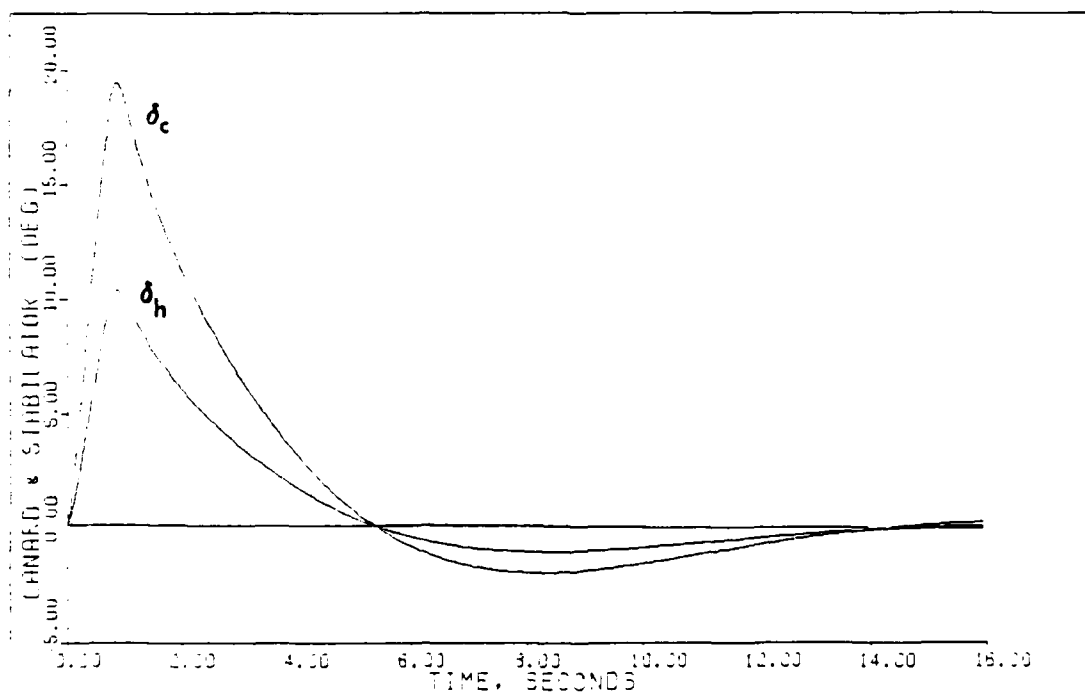
Notes:

1. See Table 5.1 for the command vector.
2. The final value of all outputs equals the commanded step input (integral control).
3. The symbol ** indicates a settling time greater than the simulation time.
4. Units for all angle outputs are in degrees, time is in seconds, and velocity in feet/second.



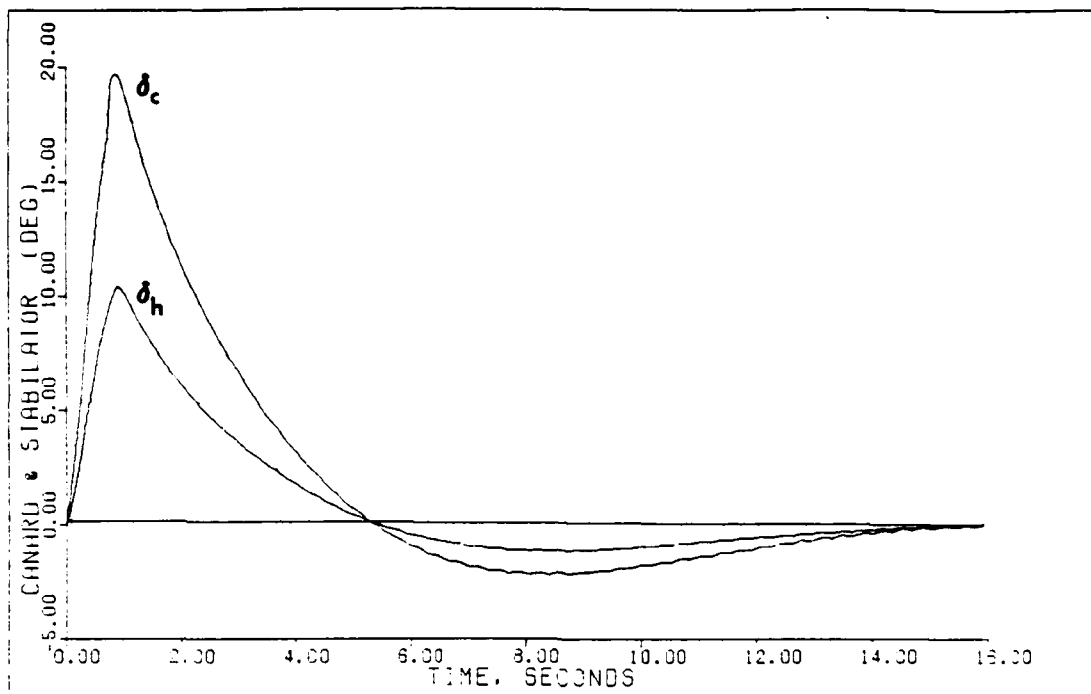
DIRECT CLIMB: BASIC PLANT (0.3M/FL200)

Fig. 5.1

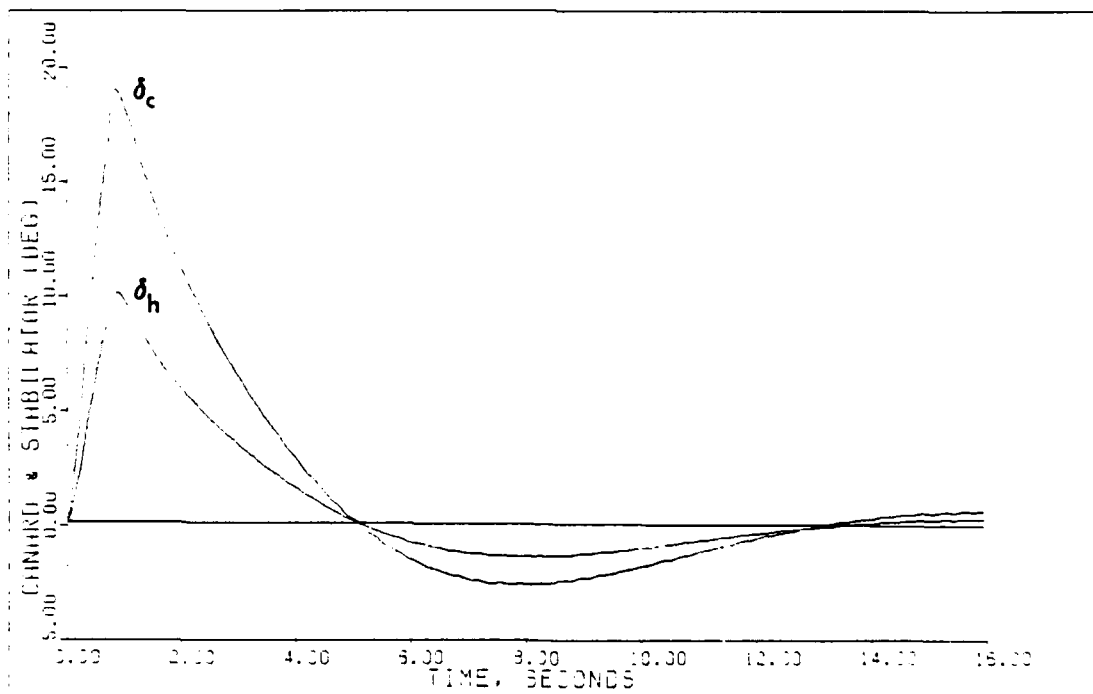


DIRECT CLIMB: PLANT+ACTUATORS (0.3M/FL200)

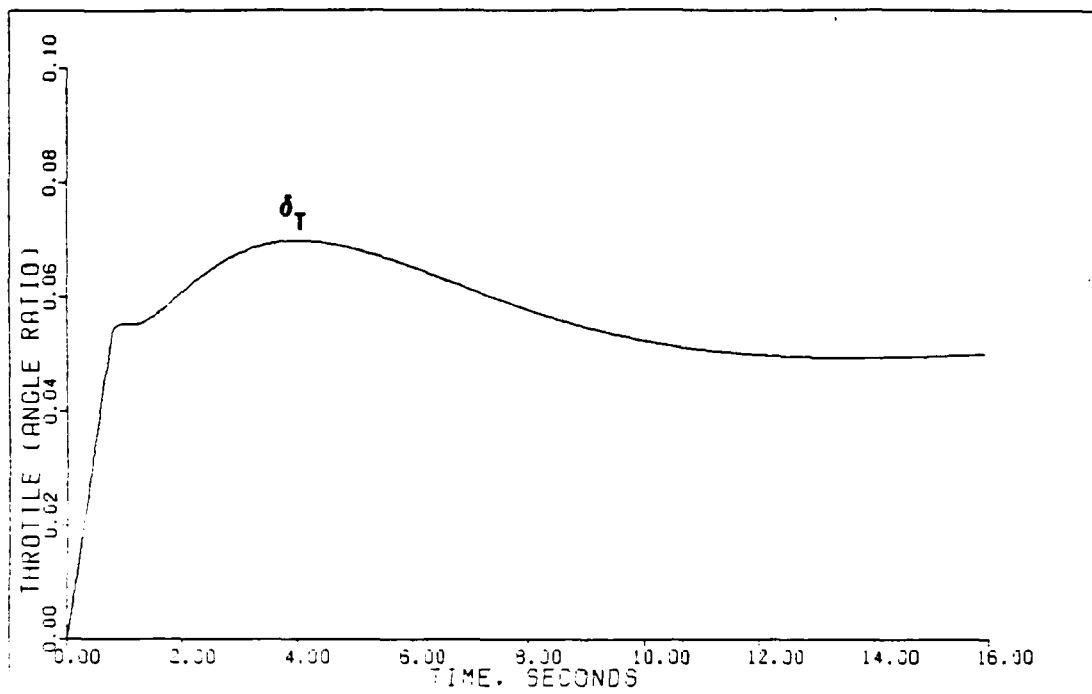
Fig. 5.2



DIRECT CLIMB: PLANT+ACTUATORS+DELAY (0.3M/FL200) **Fig. 5.3**

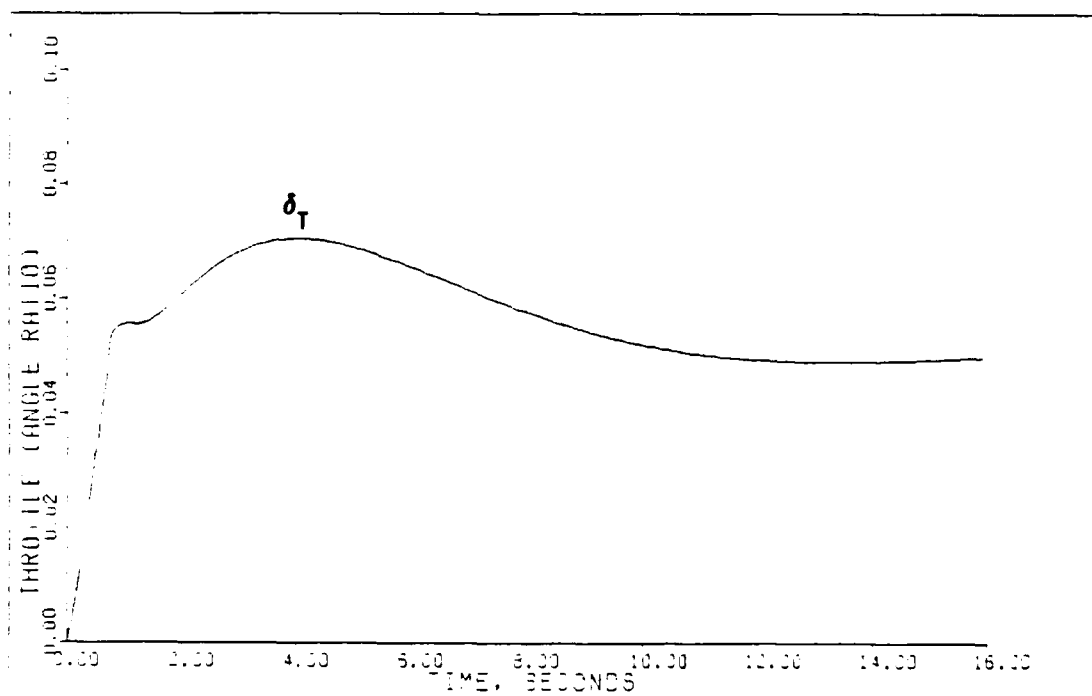


DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSORS (0.3M/FL200) **Fig. 5.4**



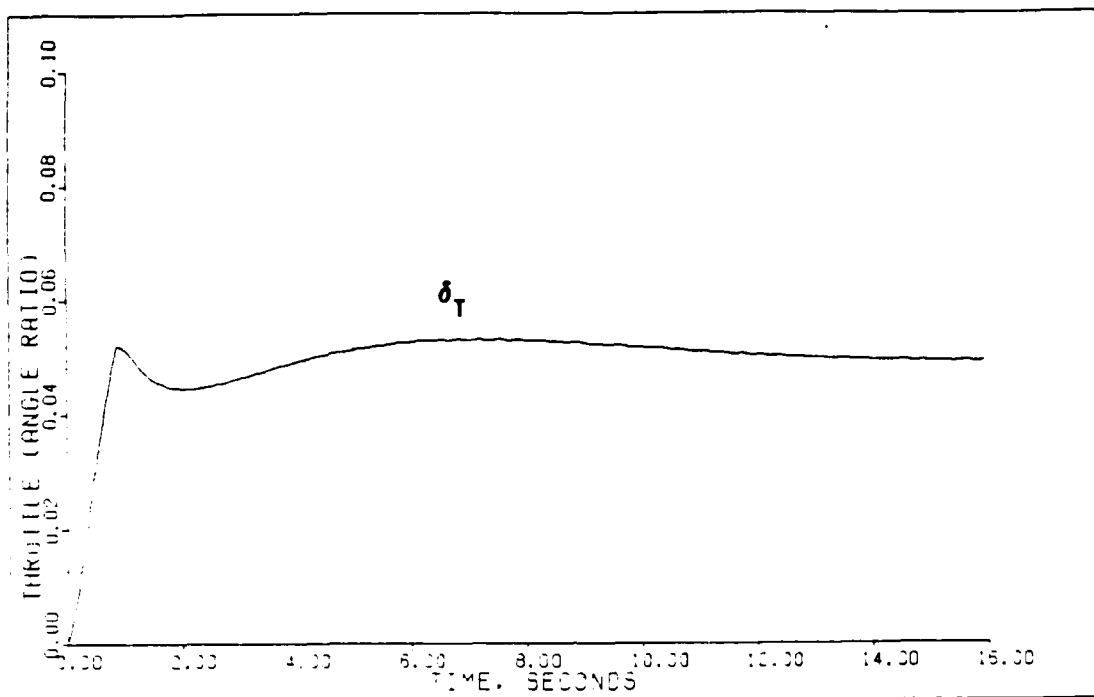
DIRECT CLIMB: BASIC PLANT (0.3M/FL200)

Fig. 5.5

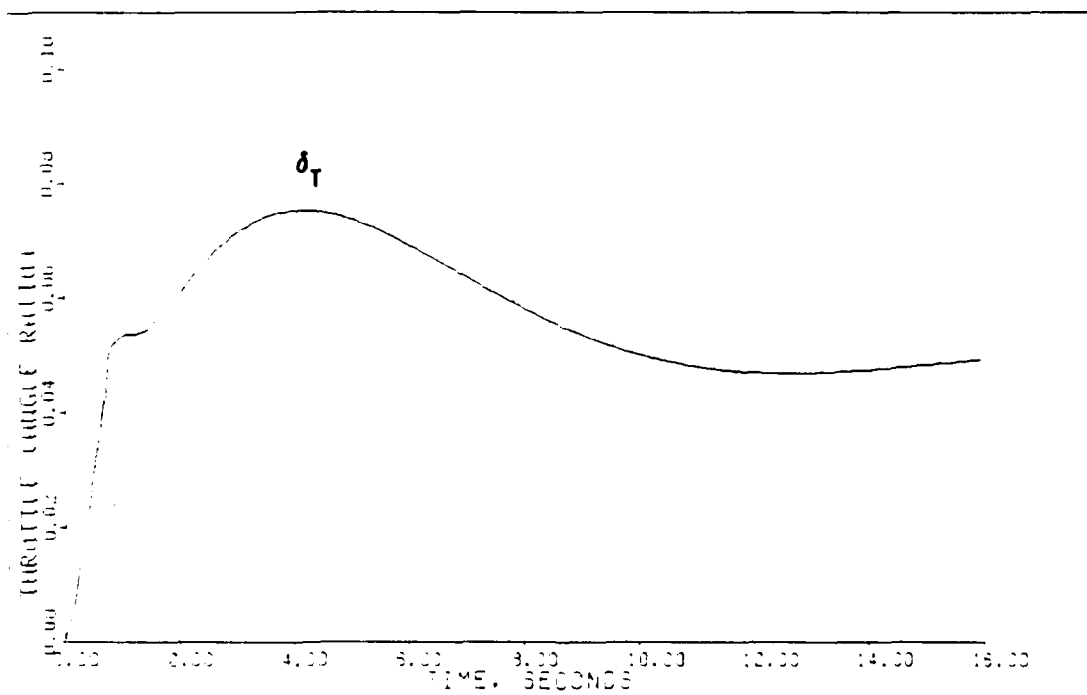


DIRECT CLIMB: PLANT-ACTUATORS (0.3M/FL200)

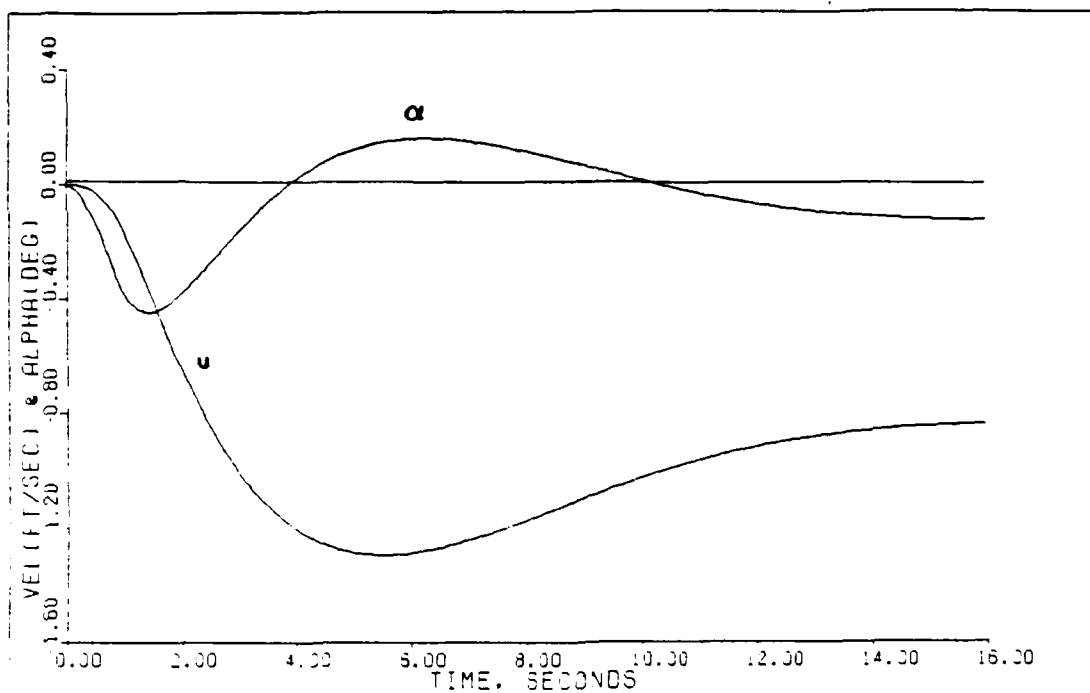
Fig. 5.6



DIRECT CLIMB: PLANT+ACTUATORS+DELAY 0.3M/F/2000 Fig. 5.7

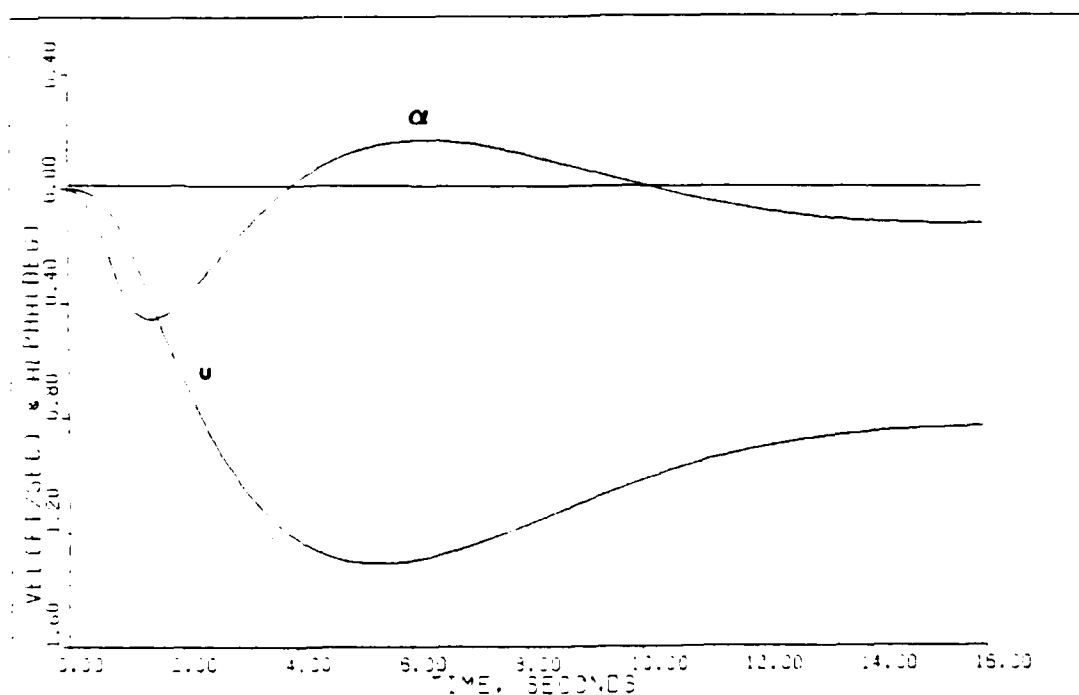


DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSORS 0.3M/F/2000 Fig. 5.8



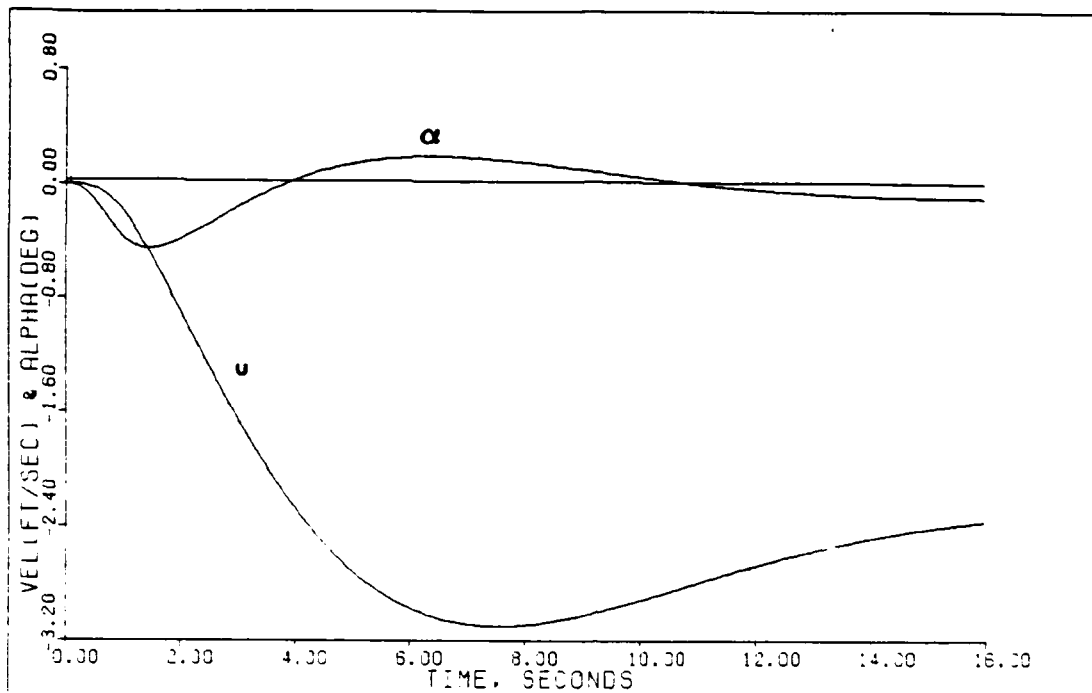
DIRECT CLIMB: BASIC PLANT (0.3M/FL000)

Fig. 5.9

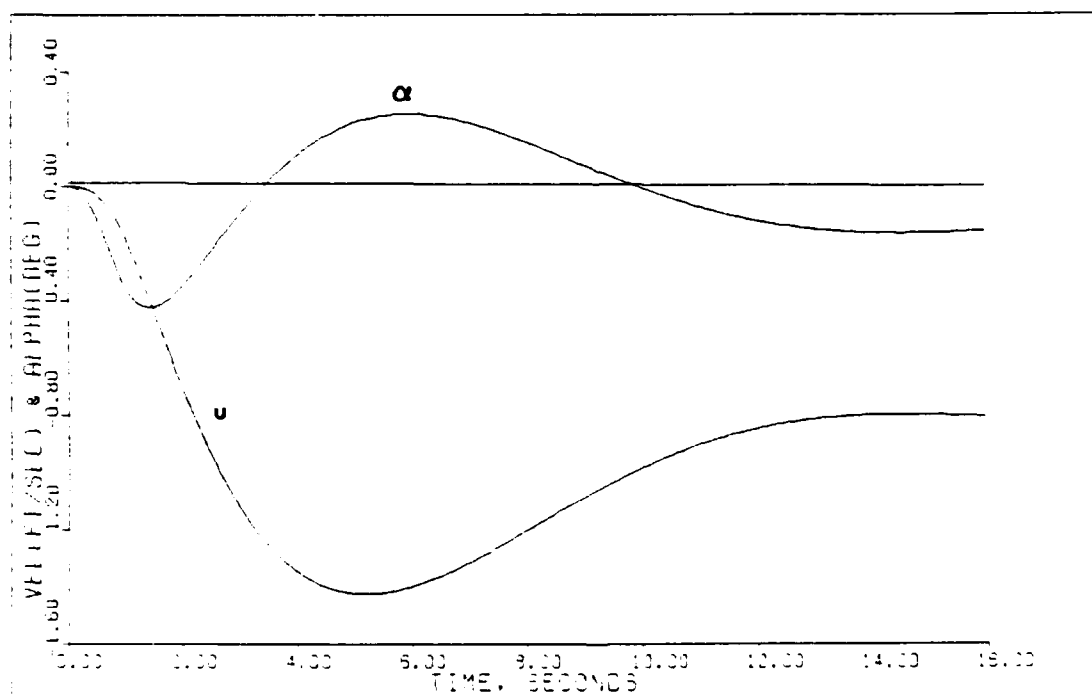


DIRECT CLIMB: PLANT-ACTUATORS (0.3M/FL000)

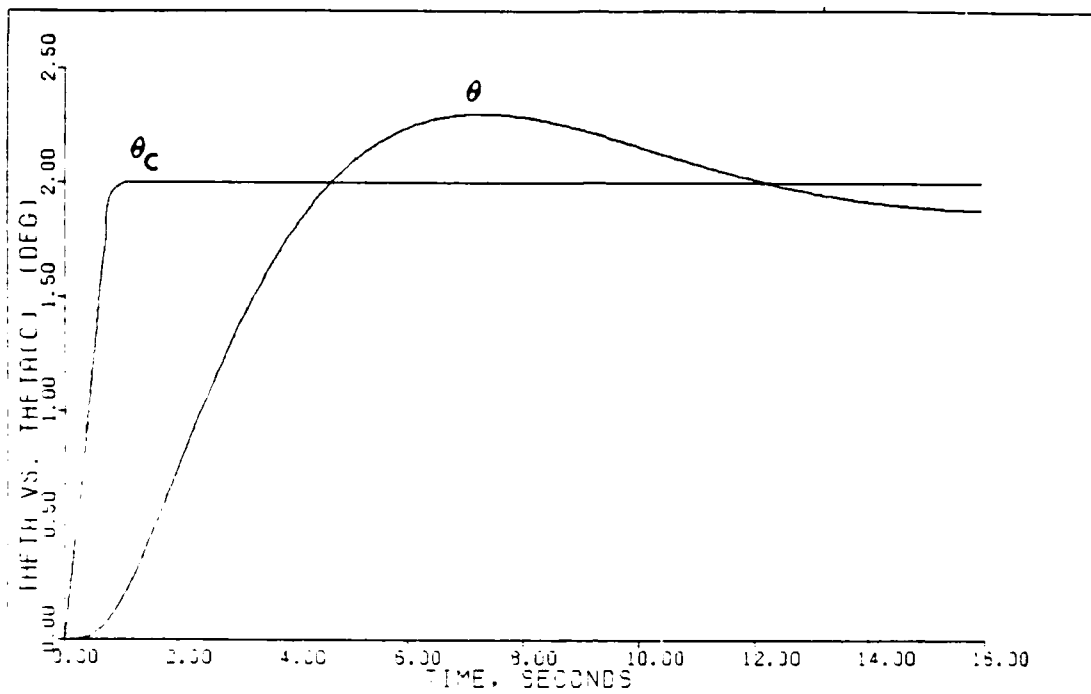
Fig. 5.10



DIRECT CLIMB: PLANT-ACTUATORS-DELAY (0.3M/FL200) Fig. 5.11

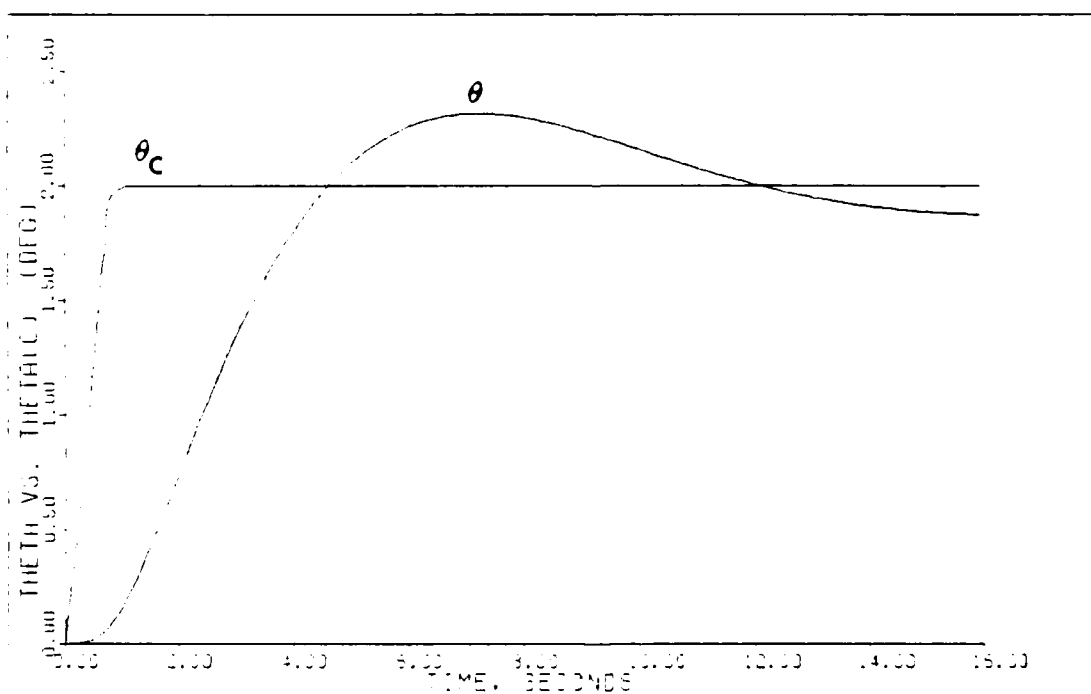


DIRECT CLIMB: PLANT-ACTUATORS-DELAY-SENSORS (0.3M FL200) Fig. 5.12



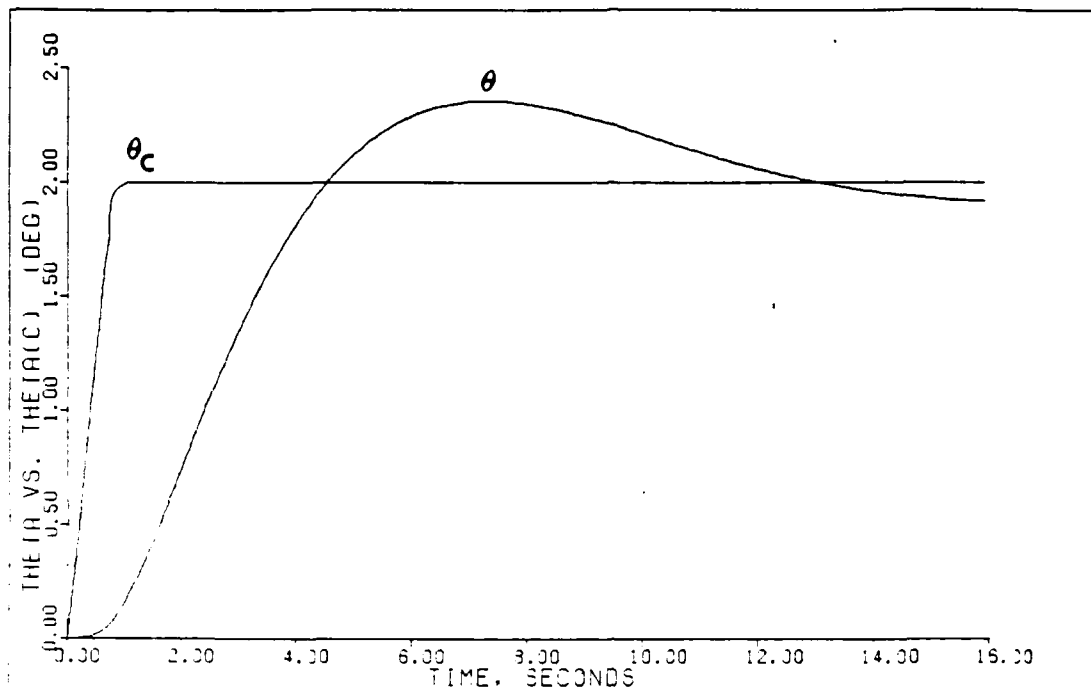
DIRECT CLIMB: BASIC PLANT (0.3M/FL200)

Fig. 5.13



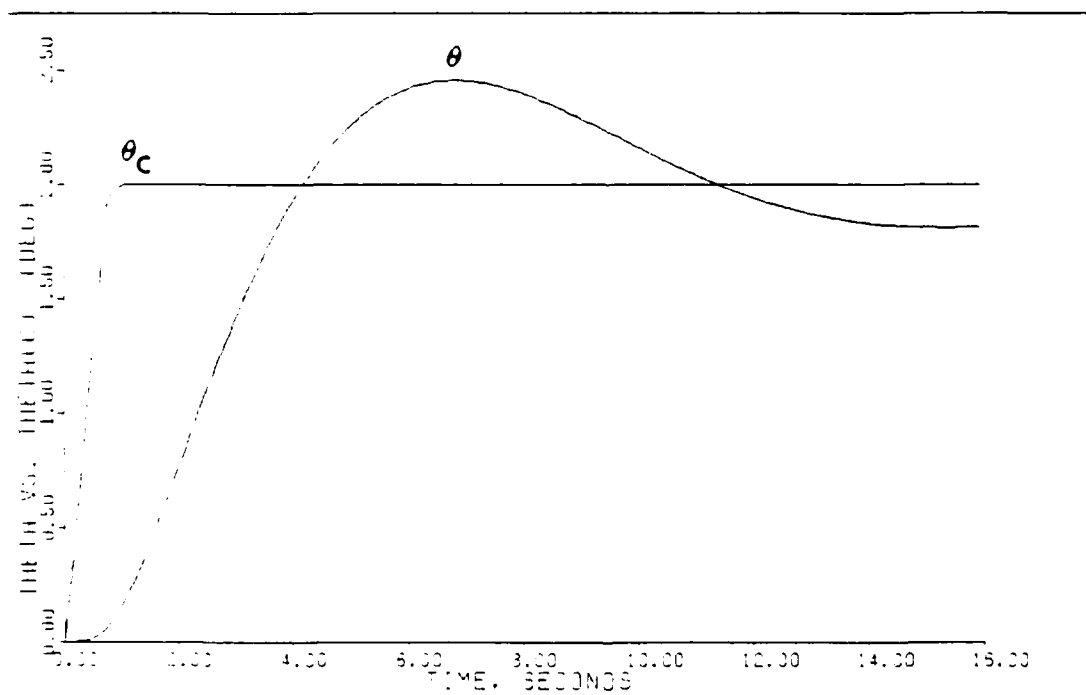
DIRECT CLIMB: PLANT+ACTUATORS (0.3M/FL200)

Fig. 5.14



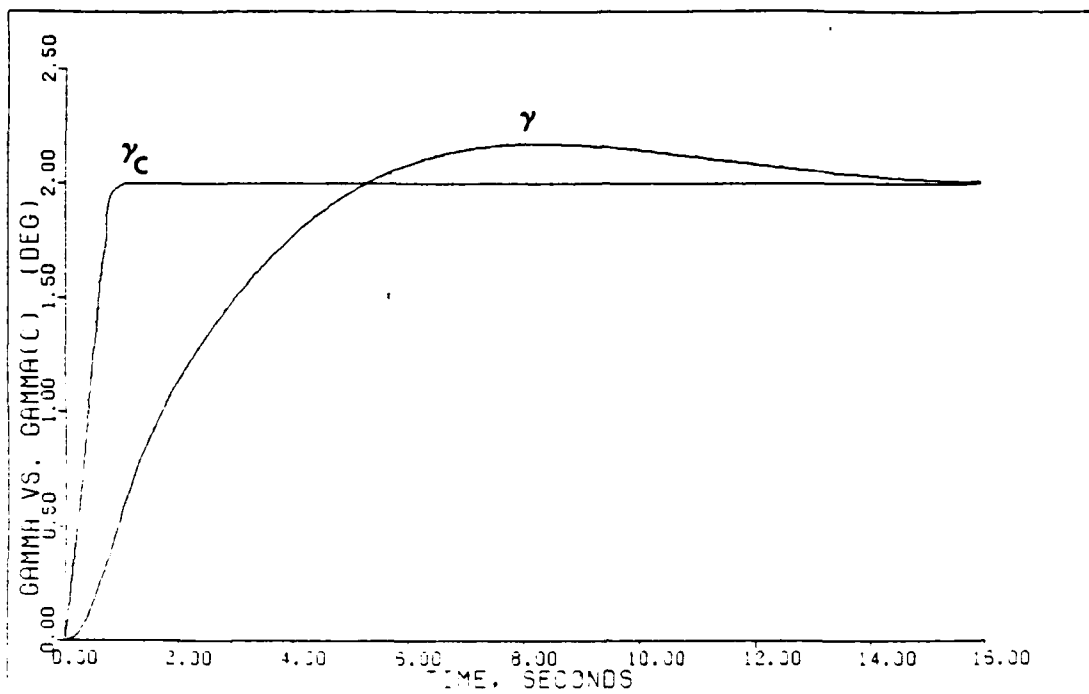
DIRECT CLIMB: PLANT+ACTUATORS-DELAY (0.3M/FLOOD)

Fig. 5.15



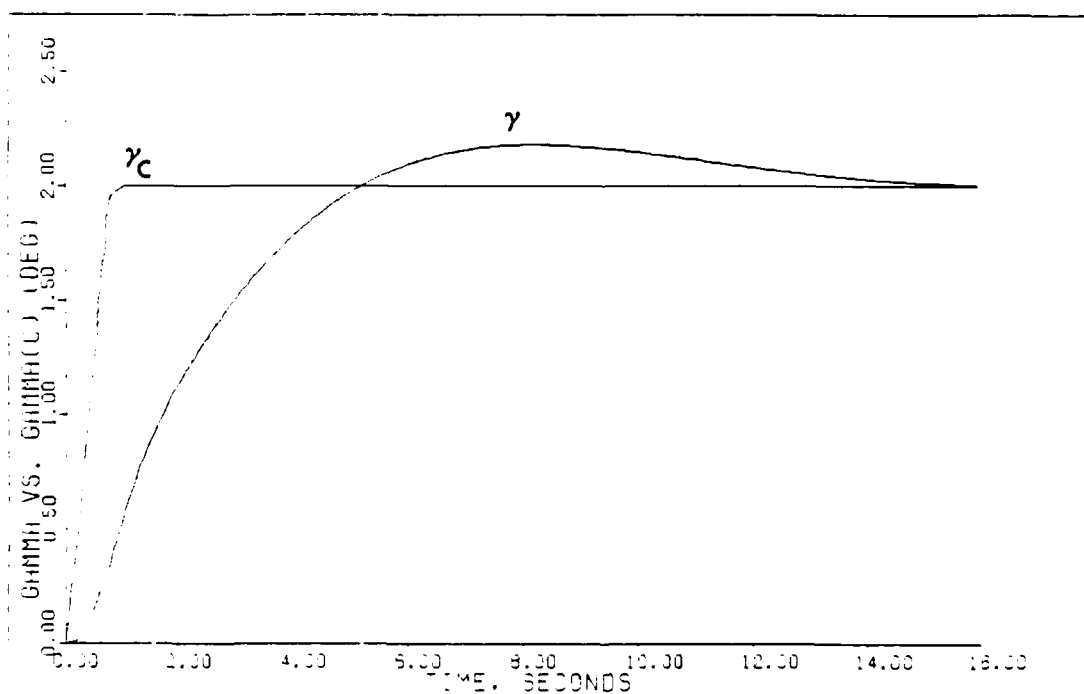
DIRECT CLIMB: PLANT+ACTUATORS-DELAY+SENSORS (0.3M/FLOOD)

Fig. 5.16



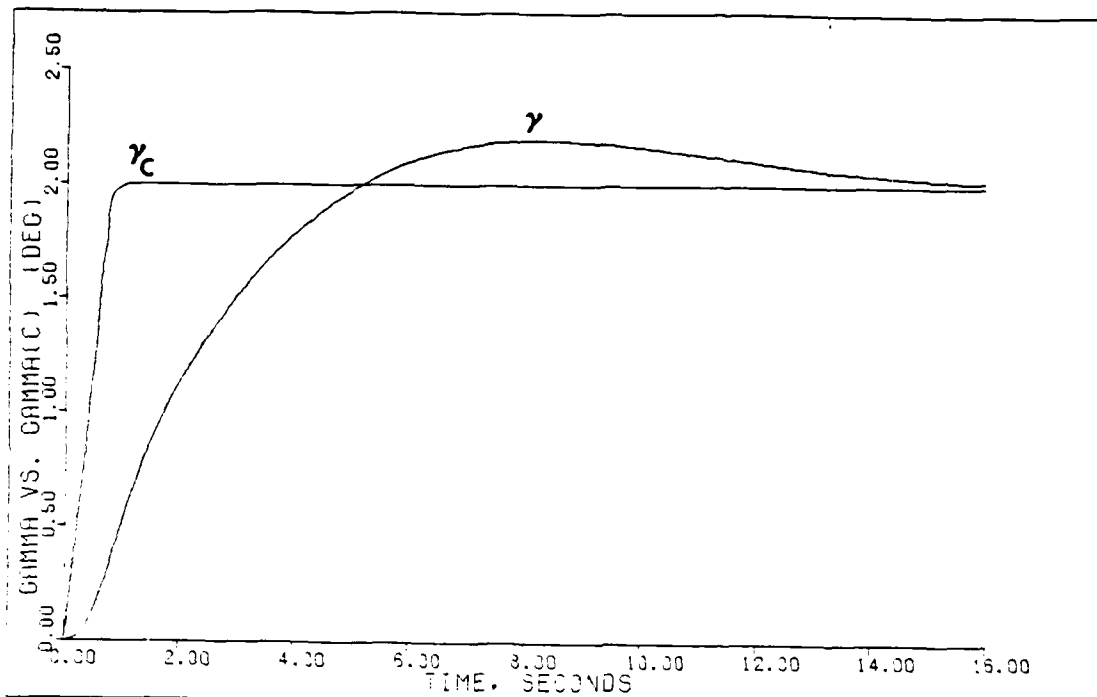
DIRECT CLIMB: BASIC PLANT (0.3M/FL200)

Fig. 5.17



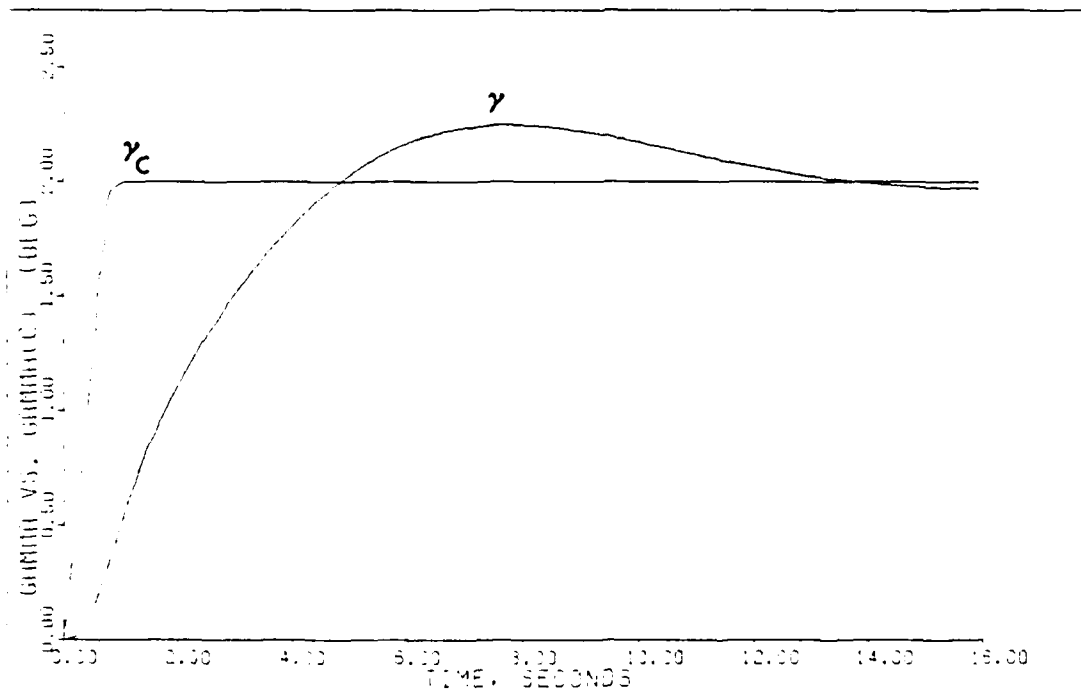
DIRECT CLIMB: PLANT-ACTUATORS (0.3M/FL200)

Fig. 5.18



DIRECT CLIMB: PLANT-ACTUATORS-DELAY (0.3M/FL200)

Fig. 5.19



DIRECT CLIMB: PLANT-ACTUATORS-DELAY-SENSORS 0.3M/FL200

Fig. 5.20

the simulation. Figures 5.3 and 5.7 give the control inputs response to the 2.0 deg commanded climb. The σ_2 value influencing velocity is 0.9 in this design as compared with 2.5 in the previous design. The decoupling effects become apparent, once again, as the peak value for velocity now reaches -3.1 ft/sec (Figure 5.11) due to its correspondingly lower gain. In addition, since the throttle's only entry in the \underline{B} matrix is in the \dot{u} equation, the throttle transient is now independently affected by σ_2 (Figure 5.7). With respect to stability, the overshoot in both theta and gamma is slightly higher as a result of the increased delay (Figures 5.15 and 5.19).

The final stage is the addition of sensor dynamics. The increased phase lag causes a slightly larger undershoot in the canard and stabilator (Figure 5.4) with a correspondingly larger overshoot in the outputs theta and gamma (Figures 5.16 and 5.20).

Overall, the system responses are smooth and well behaved as the complexity of the model is increased. At the flight conditions where dynamic pressure is greater, higher integral gains are used which results in shorter output settling times (Appendix D).

5.3 Vertical Translation (0.9 Mach/FL 200)

Vertical translation is another maneuver which demonstrates the decoupled behavior of the Porter method

multivariable controller design. While maintaining a constant pitch attitude (the pilot's forward view remains unchanged), the use of direct force is used to control flight path in a decoupled manner. At this flight condition, the aircraft is commanded to vertically translate at a flight path angle of +1.8 degs which is approximately equivalent to +1620 ft/min vertical velocity. The canard, as in the direct climb, is the control input which limits the maximum flight path commanded.

Figures 5.21 and 5.25 present the control input response of the basic plant to the commanded maneuver. Basically, the speed of response is limited by the maximum canard deflection rate (23 deg/sec). The responses are smooth with rapid rise times and minimal overshoot, characteristic of a high gain, error-actuated controller. Figures 5.29 and 5.33 give the output responses of the basic plant. The desired flight path of +1.8 degs is reached in 3.1 secs with a smooth, deadbeat response. Both gamma and velocity perturbations are held to a minimum and both transients settle to zero within 4 secs. The figures of merit for the output responses are found in Table 5.4.

During this design, the σ_3 parameter, which affects the flight path output, was very effective in controlling the response time of the canard. Its value of 0.05 was necessary to prevent canard overshoot in the transient

portion of the response. The drawback to this result is that the lower gain on the flight path channel allows for a slower response in gamma. This slower response, however, is essentially optimal for this model since the control inputs cannot respond any faster due to their rate limitations. This result is independent of any design method.

As actuators are added, the system responses are basically unaffected. Figures 5.22 and 5.26 present the control input responses which are nearly identical to those of the basic plant. With the addition of the actuators, there are no modifications made to the design parameters. Figures 5.30 and 5.34 give the output responses which again, are identical to the basic model.

The addition of computational time delay significantly affects the system response. Figures 5.23 and 5.27 show the effects of the increased delay on the control inputs. Even with a reduction in overall gain (smaller value of epsilon), there is a slight instability evident beyond the 14 sec point in the simulation. Since this is not the final design, this instability is allowed to remain for the purpose of illustration. A further reduction in gain would eliminate this problem. The net effect of the increased delay is a slower responding system (Figures 5.31 and 5.35). The settling time for gamma is now 5.9 secs with slightly larger peak values for both velocity and pitch angle. The outputs are still smooth and well behaved,

TABLE 5.3

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Vertical Translation (+1.8 degs)

Flt Condition: 0.9 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 0, 0, 0, 0$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0.8, 0.03142, 20, 20$

Basic Plant

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
		1.0	.8922E+01	.0000E+00	.1404E+02
1.429	0.770	1.0	-.2152E+01	.0000E+00	.8357E+01
		0.5	-.1327E-01	.1711E-01	.2455E-01

Plant + Actuators

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
		1.0	.8922E+01	.0000E+00	.1404E+02
1.429	0.770	1.0	-.2152E+01	.0000E+00	.8357E+01
		0.05	-.1327E-01	.1711E-01	.2455E-01

Plant + Actuators + Delay

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
		1.0	.7705E+01	.0000E+00	.1213E+02
1.429	0.665	0.8	-.1859E+01	.0000E+00	.7217E+01
		0.05	-.1146E-01	.1182E-01	.2120E-01

Plant + Actuators + Delay + Sensors

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
		1.0	.6894E+01	.0000E+00	.1085E+02
1.429	0.595	0.5	-.1663E+01	.0000E+00	.6458E+01
		0.05	-.1026E-01	.6611E-02	.1897E-01

Notes:

- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\alpha)\underline{K}_0$.
- Irregular Design: $\underline{M} = (0.3, 0, 0)^T$.

TABLE 5.4

DESIGN OUTPUT FIGURES OF MERIT

Maneuver: Vertical Translation (+1.8 degs)

Flt Condition: 0.9 Mach at FL 200

Basic Plant

<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	-.0636	1.575	3.8
Velocity	-.0247	0.875	3.0
Flight Path Angle	+1.804	5.250	3.15

Plant + Actuators

<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	-.0654	1.400	3.9
Velocity	-.0263	0.875	4.1
Flight Path Angle	+1.804	5.075	3.15

Plant + Actuators + Delay

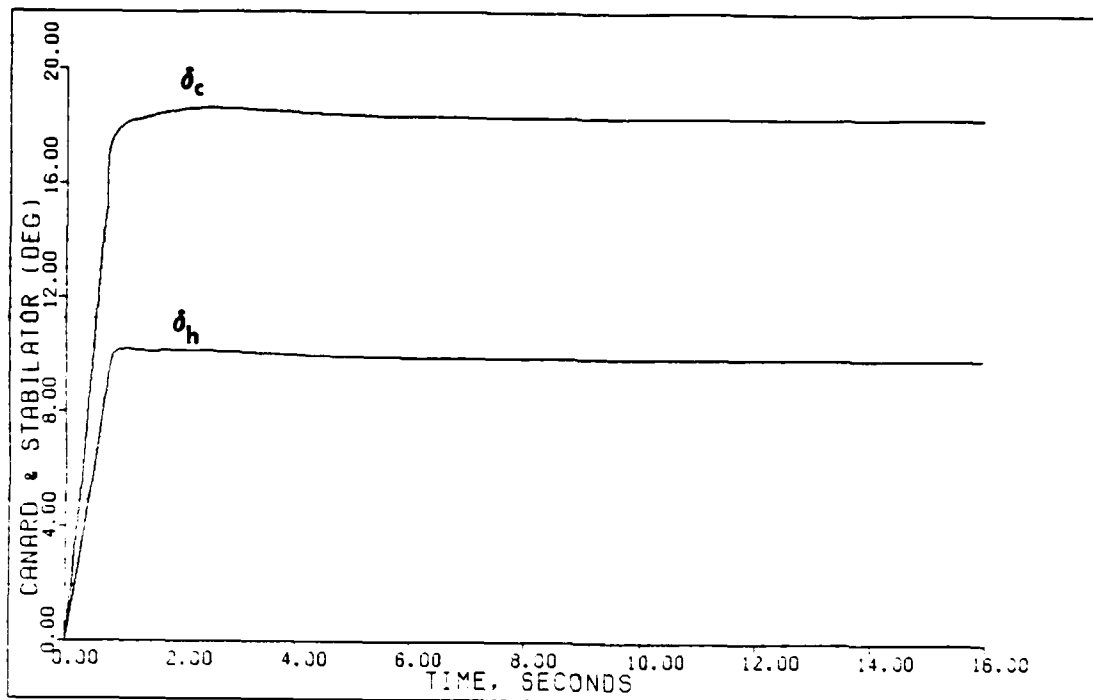
<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	-.1045	1.750	5.90
Velocity	-.0447	1.050	5.70
Flight Path Angle	+1.801	9.975	5.95

Plant + Actuators + Delay + Sensors

<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	-.1206	1.575	6.20
Velocity	-.0880	0.875	6.10
Flight Path Angle	+1.801	10.15	6.12

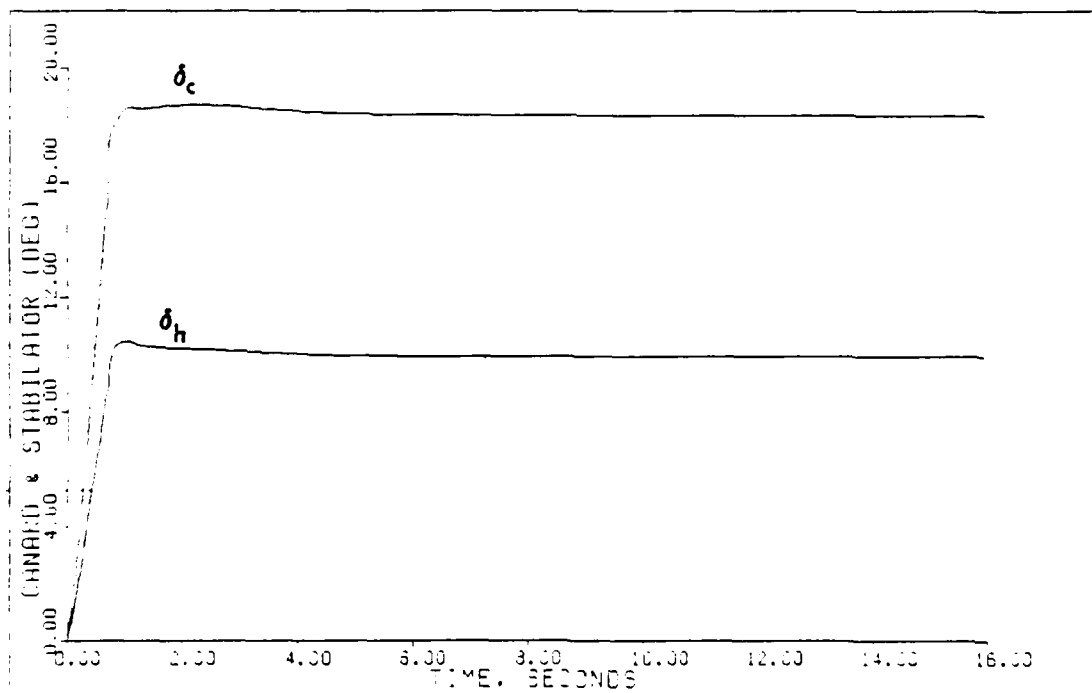
Notes:

1. See Table 5.3 for the command vector.
2. The final value of all outputs equals the commanded step input (integral control).
3. The settling times for pitch angle and velocity are estimated from the response plots.
4. Units for all angle outputs are in degrees, time is in seconds, and velocity is in feet/second.



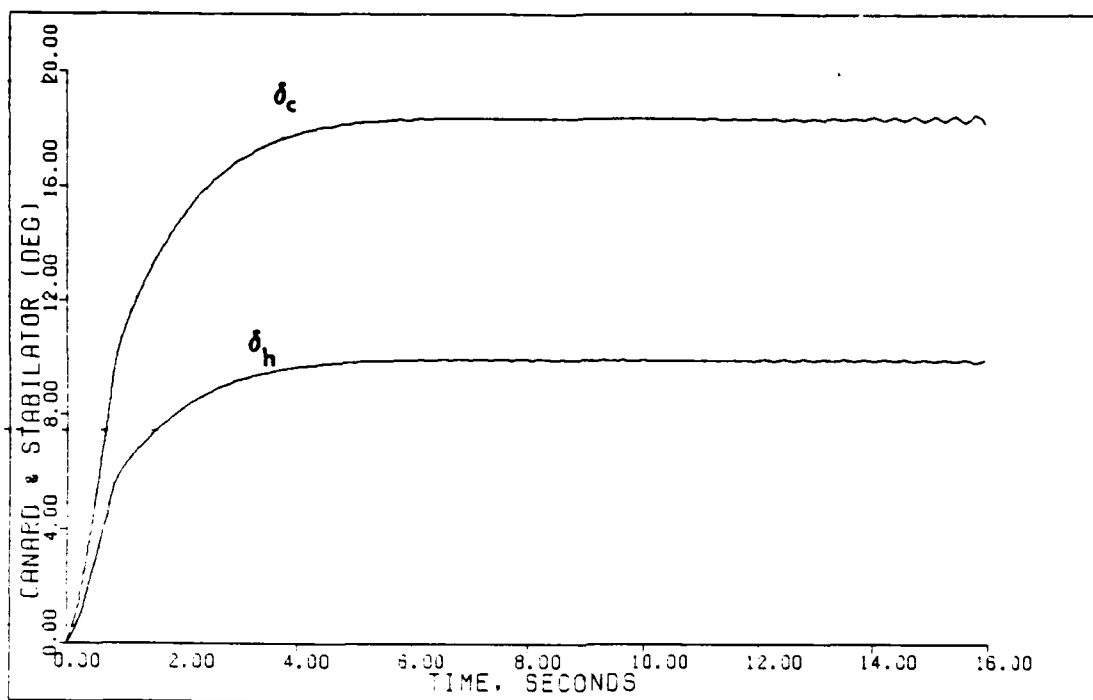
VERTICAL TRANS: BASIC PLANT (0.9M/FL200)

Fig. 5.21



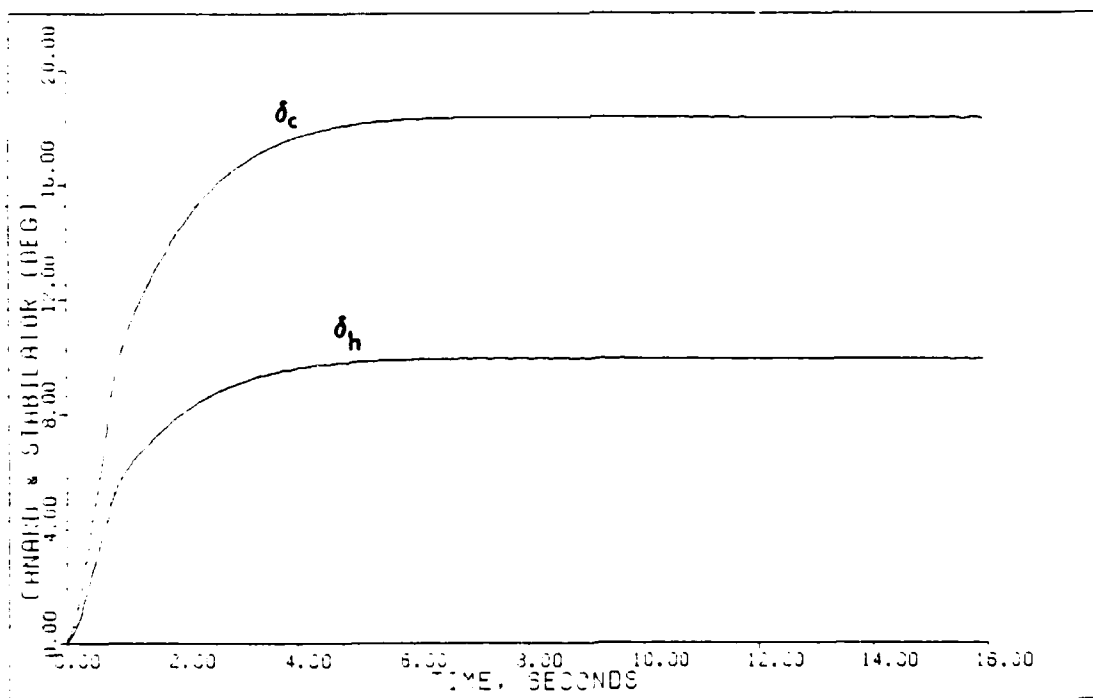
VERTICAL TRANS: PLANT-ACTUATORS (0.9M/FL200)

Fig. 5.22



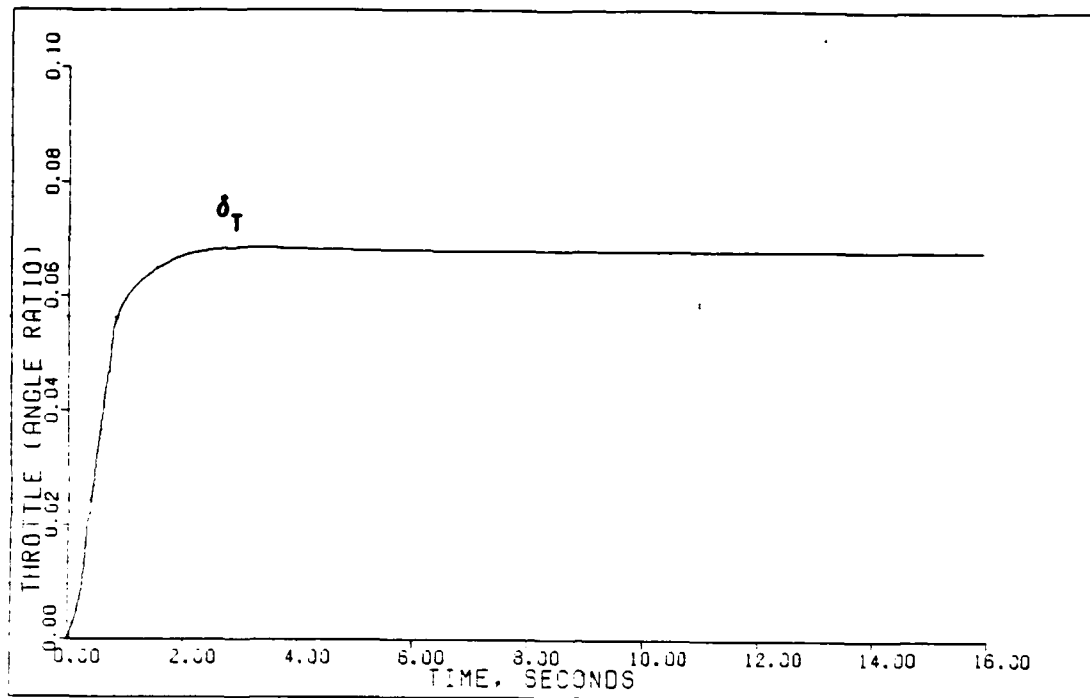
VERTICAL TRANS: PLANT-ACTUATORS+DELAY (0.9M/FL200)

Fig. 5.23



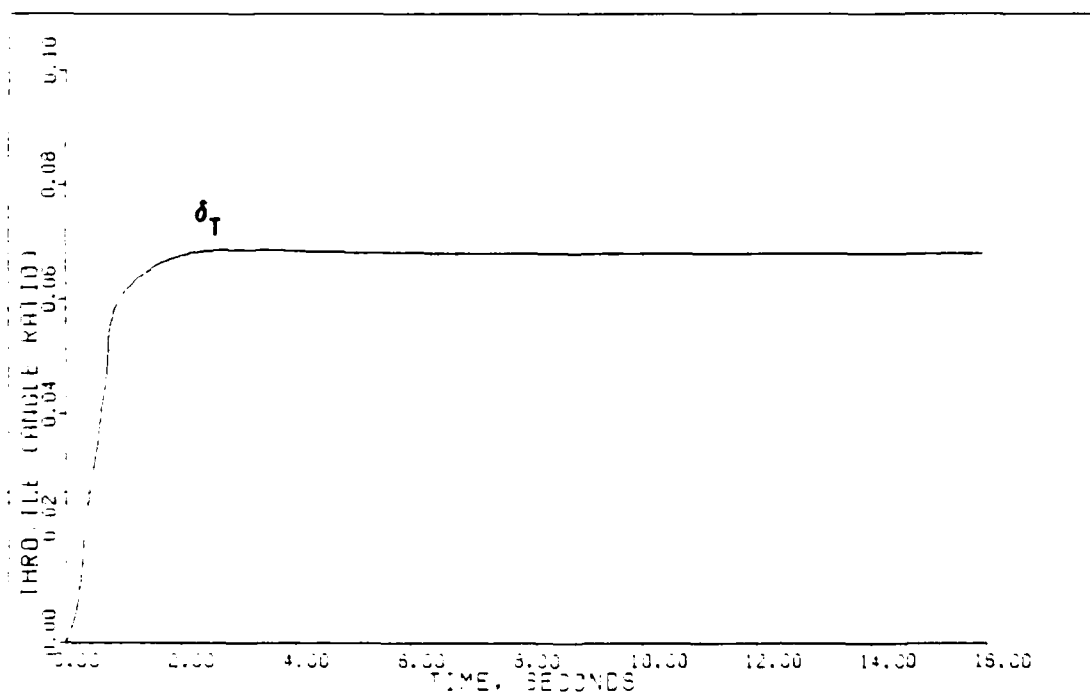
VERTICAL TRANS: PLANT-ACTUATORS-DELAY-SENSORS (0.9M/FL200)

Fig. 5.24



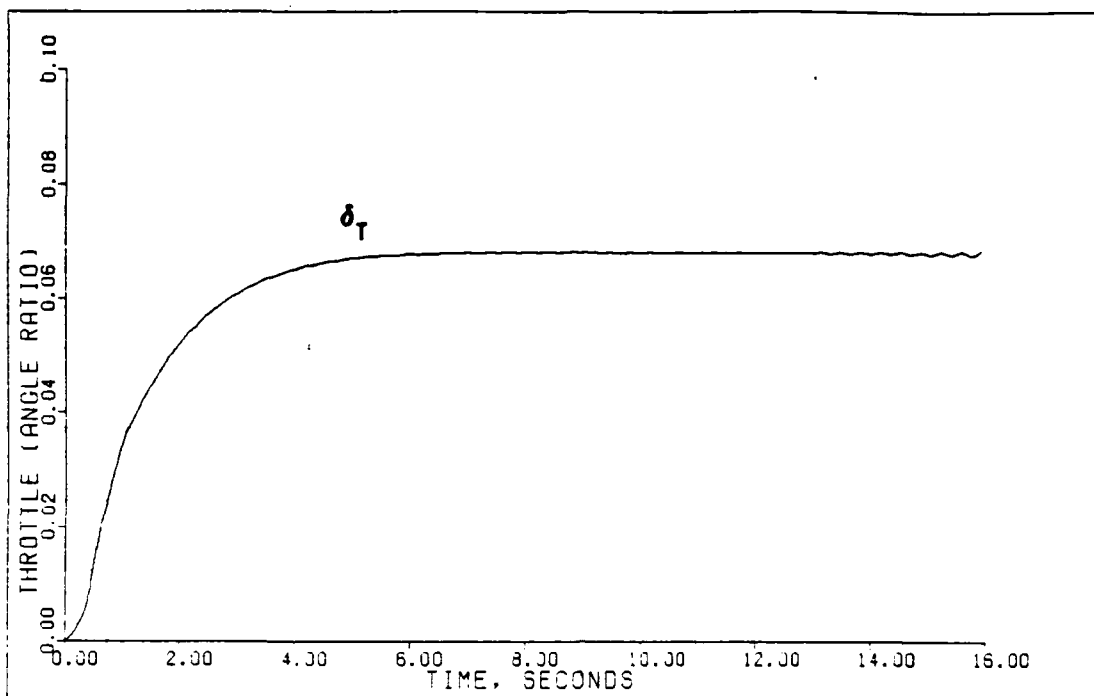
VERTICAL TRANS: BASIC PLANT (0.9M/FL200)

Fig. 5.25

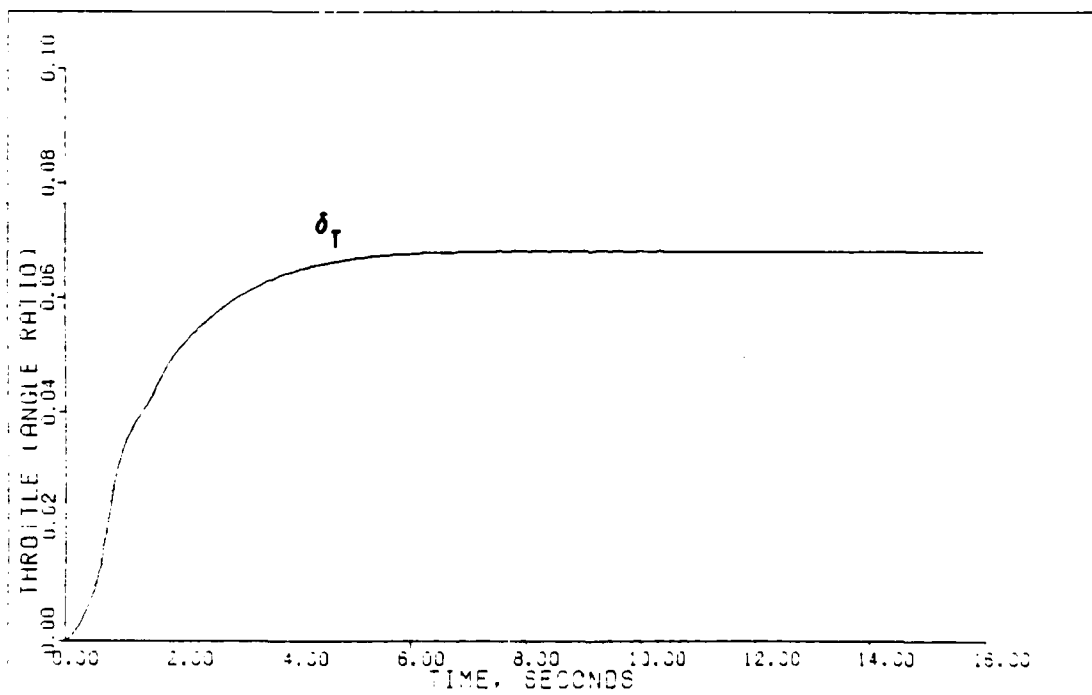


VERTICAL TRANS: PLANT-ACTUATORS (0.9M/FL200)

Fig. 5.26

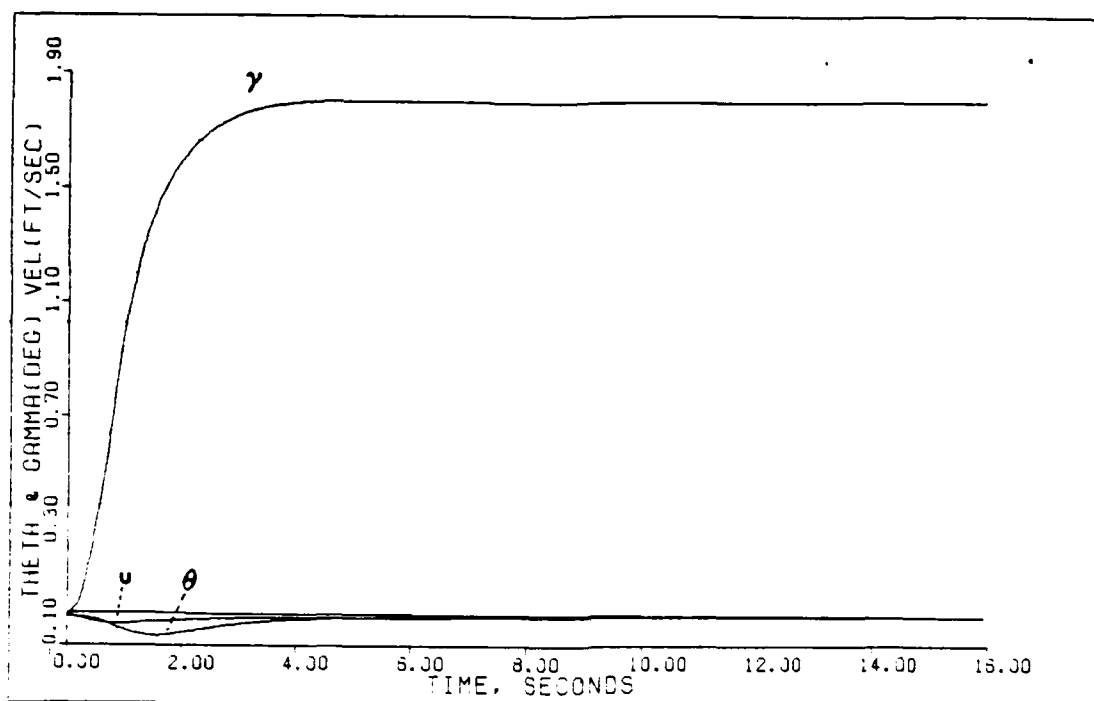


VERTICAL TRANS: PLANT-ACTUATORS+DELAY (0.3M/FL200) Fig. 5.27



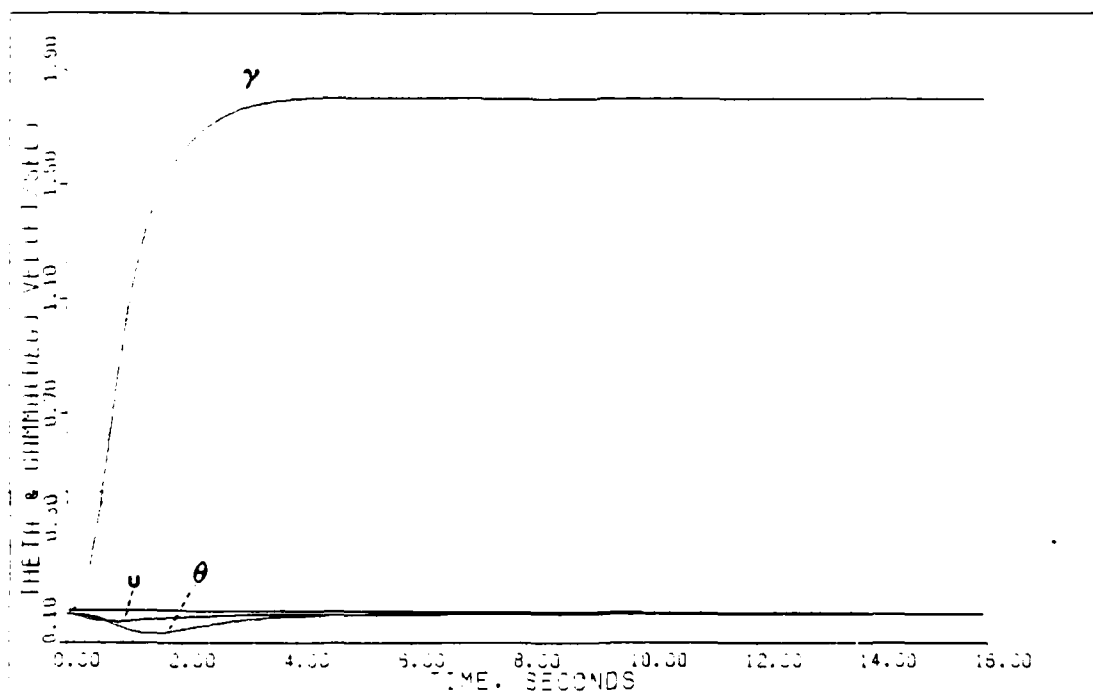
VERTICAL TRANS: PLANT-ACTUATORS+DELAY+SENSORS (0.3M/FL200)

Fig. 5.28



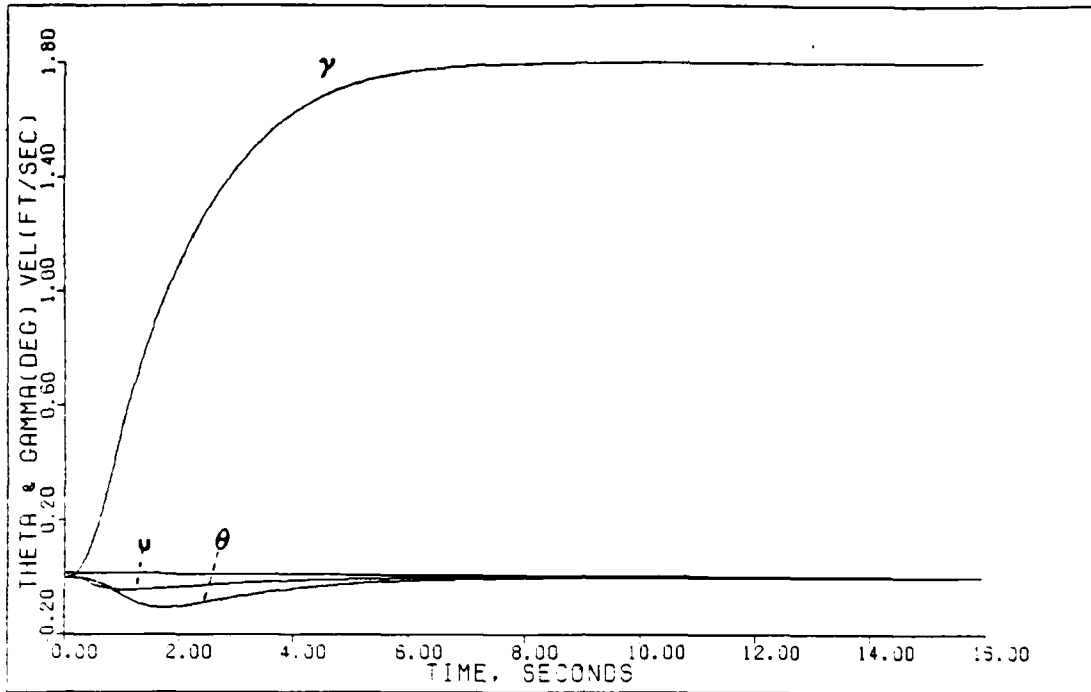
VERTICAL TRANS: BASIC PLANT (0.9M/FL200)

Fig. 5.29

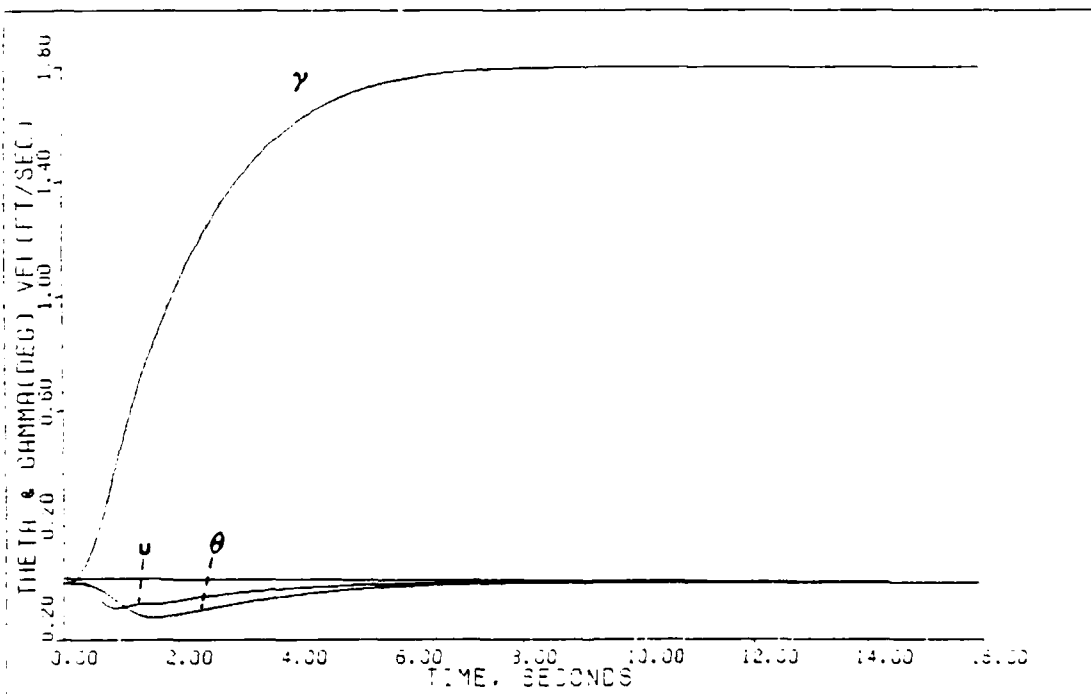


VERTICAL TRANS: PLANT-POTULATORS (0.9M/FL200)

Fig. 5.30

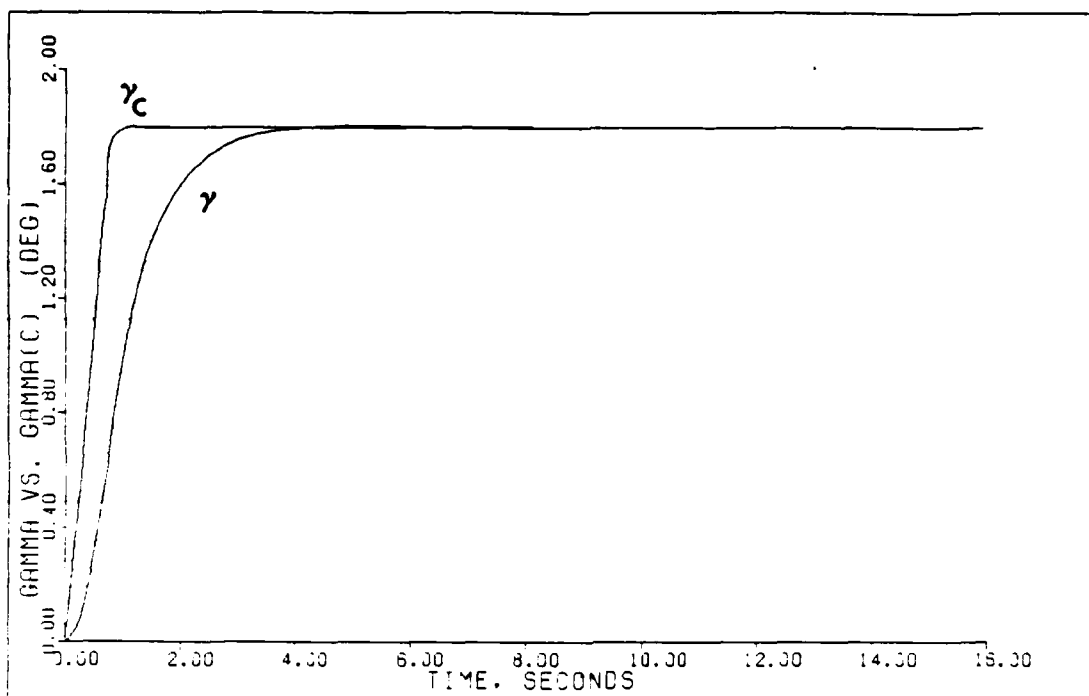


VERTICAL TRANS: PLANT+ACTUATORS-DELAY (0.9M/FL200) Fig. 5.31



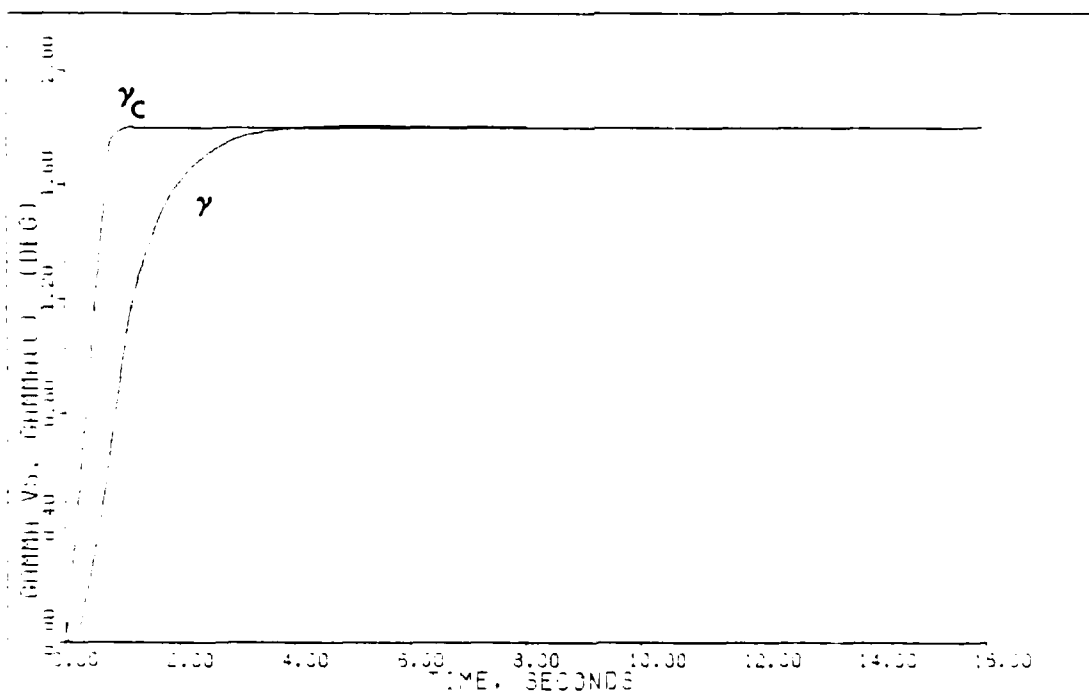
VERTICAL TRANS: PLANT+ACTUATORS-DELAY+SENSORS 0.9M, FL200

Fig. 5.32



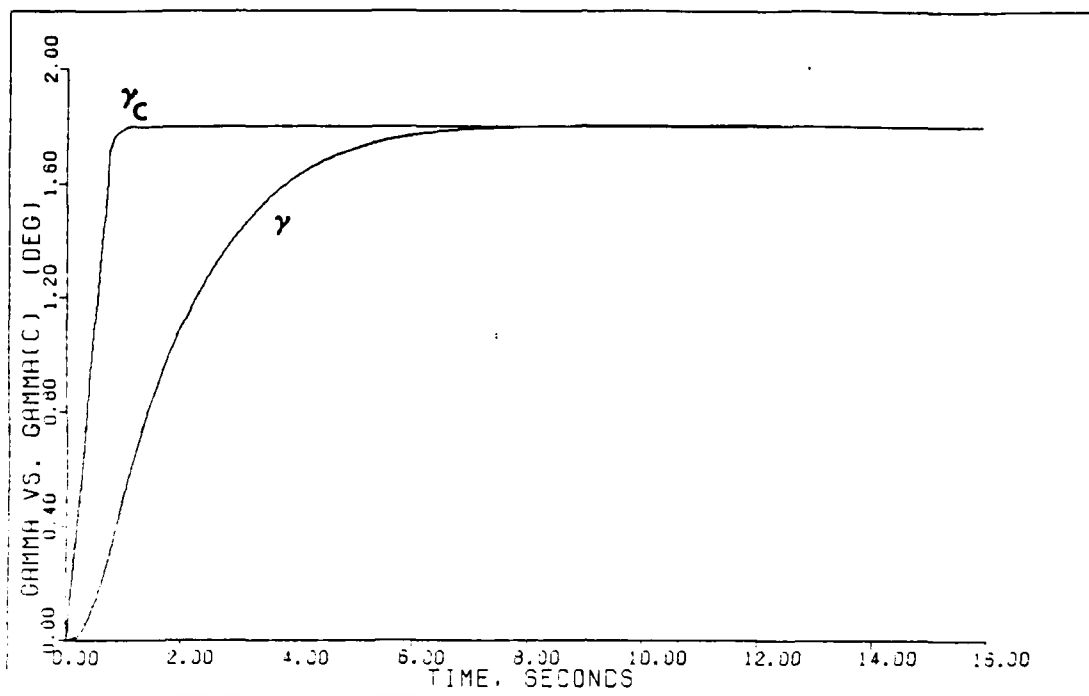
VERTICAL TRANS: BASIC PLANT (0.9M/FL200)

Fig. 5.33

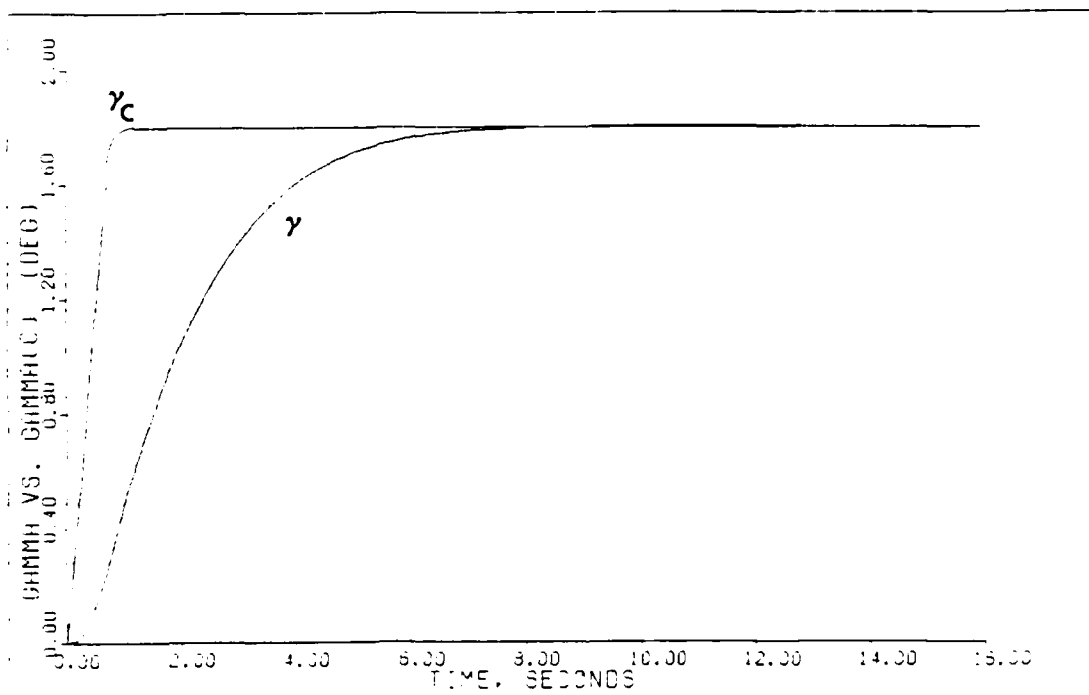


VERTICAL TRANS: PLANT-ACTUATORS (0.9M/FL200)

Fig. 5.34



VERTICAL TRANS: PLANT+ACTUATORS+DELAY (0.9M/FL200) Fig. 5.35



VERTICAL TRANS: PLANT+ACTUATORS+DELAY+SENSORS (0.9M/FL200) Fig. 5.36

despite the slower response. It is interesting to note that there is no instability present in any of the outputs as a result of the input oscillations beyond 14 secs. This is indicative of the systems lower bandwidth and is more graphically shown later in this chapter under noise effects.

The final addition of sensor dynamics results in smooth and stable control inputs after modifications to the design parameters (Figures 5.24 and 5.28). After a reduction in gain (Table 5.3), the control inputs demonstrate stability throughout the simulation time with essentially the same characteristics as the previous design. Σ_2 is also reduced to prevent instability in the throttle input. The output responses for the fully developed model are given in Figures 5.32 and 5.36. The settling time for flight path has now increased to 6.1 secs with a slight increase in the velocity peak due to the lower sigma gain.

Overall, the maneuver results in a smooth increase in flight path to +1.8 degs with negligible changes in velocity and pitch angle. Appendix D includes the results at each of the other three flight conditions.

5.4 Pitch Pointing (1.4 Mach/FL 200)

Pitch pointing is a maneuver that allows the pilot the capability to independently control pitch attitude (θ) without changing the flight path or equilibrium velocity of the aircraft. Once again, the use of direct

lift allows the aircraft to produce a pitching moment that increases θ while the canard and stabilator work together to balance forces in the z-axis and keep the perturbation in γ near zero. The canard, once more, is the limiting control input at this particular flight condition. The maximum pitch pointing capability, irrespective of design methods, is +2.1 degs at 1.4 Mach/FL 200.

Figures 5.37 and 5.41 present the control input deflections for the basic aircraft. The canard moves smoothly at near maximum rate and consequently limits both the speed and magnitude of the output responses for this maneuver. The initial throttle response is negative which, at first glance, appears to be an error. Upon investigation of the trimmed canard and stabilator values (Appendix C), the answer becomes obvious. The stabilator ($\alpha_{ts} = 3.06$ degs) begins a negative deflection but is actually decreasing total drag since it is trimmed at a positive value. The canard, however, increases drag as it deflects upward (negative) since its trimmed value is already negative ($\alpha_{ts} = -2.18$ degs). These two effects seem to offset each other, except that the stabilator control derivative in the velocity equation is approximately 20 times more effective than the canard. As a result, the stabilator dominates and overall aircraft drag is momentarily reduced.

Figures 5.45 and 5.49 depict the output responses to the pitch pointing command. Theta settles to the commanded value in 4.2 secs with flight path peaking at a negligible 0.038 degs. Velocity begins to decrease with the reduced throttle and increasing induced drag. This condition is quickly compensated by a rapidly advancing throttle input. The σ_2 value is reduced below unity to maintain stability in the throttle input. The lower sigma gain value allows for "looser" control of the output variable, u . This result is still quite acceptable since the peak transient is only -0.18 ft/sec and settles within 4 secs.

Actuator dynamics affect only the throttle input and its associated output, velocity. Figures 5.42 and 5.46 show the actuators affect on the throttle which results in a slightly more active velocity response. The peak value of -0.185 ft/sec is insignificant and does not detract from the overall performance.

Computational time delay tends to destabilize the control inputs after 14 secs of simulation (Figure 5.39). The ripple is most apparent in the canard. Since this is not the final design, this instability is not critical. The most dramatic effect is seen in the throttle input (Figure 5.43). The increased delay creates an out-of-phase condition between the throttle and the output velocity. This oscillation can be diminished by a reduction

in the σ_2 gain but the penalty paid is a larger velocity transient. Too large a transient reduces the validity of the linearized model. Reduction of this oscillation may be necessary to prevent engine damage that results from this type of control. The pitch angle is basically unaffected by the increased delay and smoothly rises to +2.1 degs within 3.85 secs.

The incorporation of sensor dynamics into the aircraft model increases the throttle oscillations (Figure 5.44). The instability problem has been eliminated with a reduction in gain ($\epsilon = 1.233$) but the relatively tight control on the velocity channel increases the oscillation in the throttle input. A reduction in gain would damp the input but result in a larger velocity transient. The velocity peak is now at -1.23 ft/sec which is well within the linearity of the model (Figure 5.48). As stated earlier, actual implementation of this design might require reduction in the thrust oscillations by relaxation of the velocity gain. Since stability is maintained and implementation is beyond the scope of this study, a "tight" control on the velocity output is chosen for the final design.

The pitch angle response is only slightly altered by the addition of sensors. Figure 5.52 compares the response of θ to its commanded input. The settling time for θ is now increased to 5.4 secs in the fully

TABLE 5.5

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Pitch Pointing (+2.1 degs)

Flt Condition: 1.4 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 0.8, 0.03665, 20, 20$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0, 0, 0, 0$

Basic Plant

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
1.111	0.999	1.2	.1335E+02	.0000E+00	.6504E+03
		0.8	-.1542E+01	.0000E+00	.2579E+03
		1.1	-.7131E-02	.6660E-02	.7470E+00

Plant + Actuators

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
1.111	0.999	1.2	.1335E+02	.0000E+00	.6504E+03
		0.8	-.1542E+01	.0000E+00	.2579E+03
		1.1	-.7131E-02	.6660E-02	.7470E+00

Plant + Actuators + Delay

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
5.000	0.270	1.2	.3608E+01	.0000E+00	.1758E+03
		0.8	-.4168E+00	.0000E+00	.6969E+02
		1.1	-.1927E-02	.1800E-02	.2019E+00

Plant + Actuators + Delay + Sensors

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
0.300	0.270	1.0	.3007E+01	.0000E+00	.1758E+03
		0.8	-.3473E+00	.0000E+00	.6969E+02
		1.1	-.1606E-02	.1800E-02	.2019E+00

Notes:

- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\alpha)\underline{K}_0$.
- Irregular Design: $\underline{M} = (0.3, 0, 0)^T$.

TABLE 5.6
DESIGN OUTPUT FIGURES OF MERIT

Maneuver: Pitch Pointing (+2.1 degs)

Flt Condition: 1.4 Mach at FL 200

Basic Plant

<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	+2.100	N/A	4.2
Velocity	-.1812	1.05	4.1
Flight Path Angle	+.0379	1.05	4.1

Plant + Actuators

<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	+2.100	N/A	4.2
Velocity	-.1851	1.05	4.1
Flight Path Angle	+.0377	1.05	4.3

Plant + Actuators + Delay

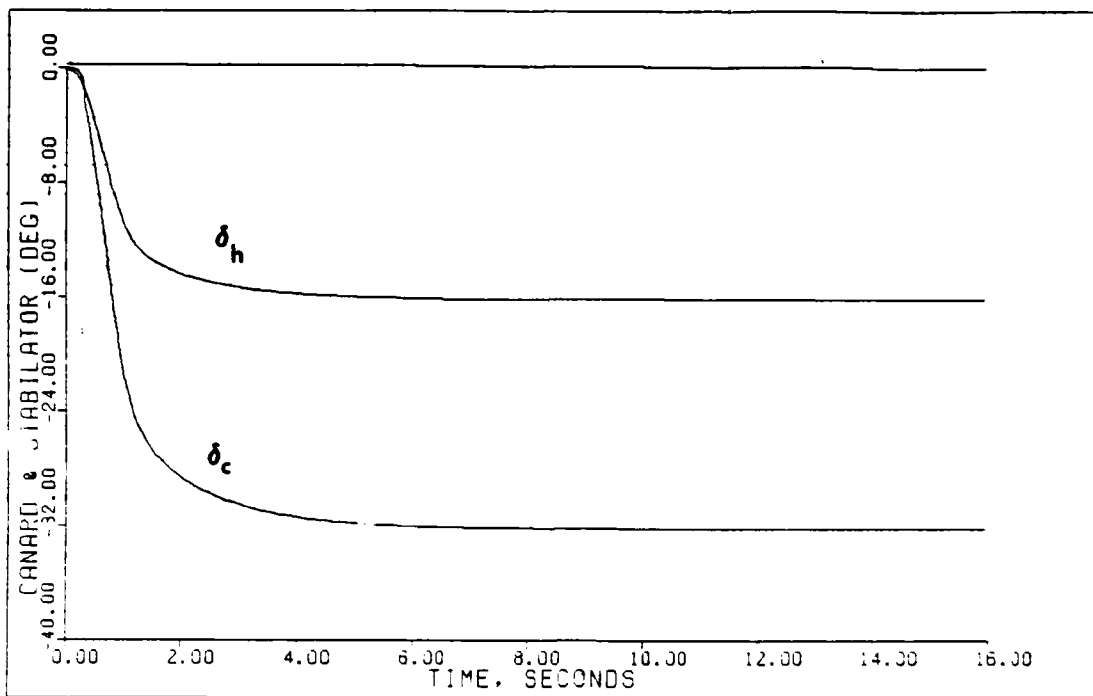
<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	+2.100	N/A	3.85
Velocity	-.8037	1.05	3.9
Flight Path Angle	+.1000	1.05	3.8

Plant + Actuators + Delay + Sensors

<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Pitch Angle	+2.100	N/A	5.42
Velocity	-1.234	1.225	6.1
Flight Path Angle	+.1390	0.875	2.1

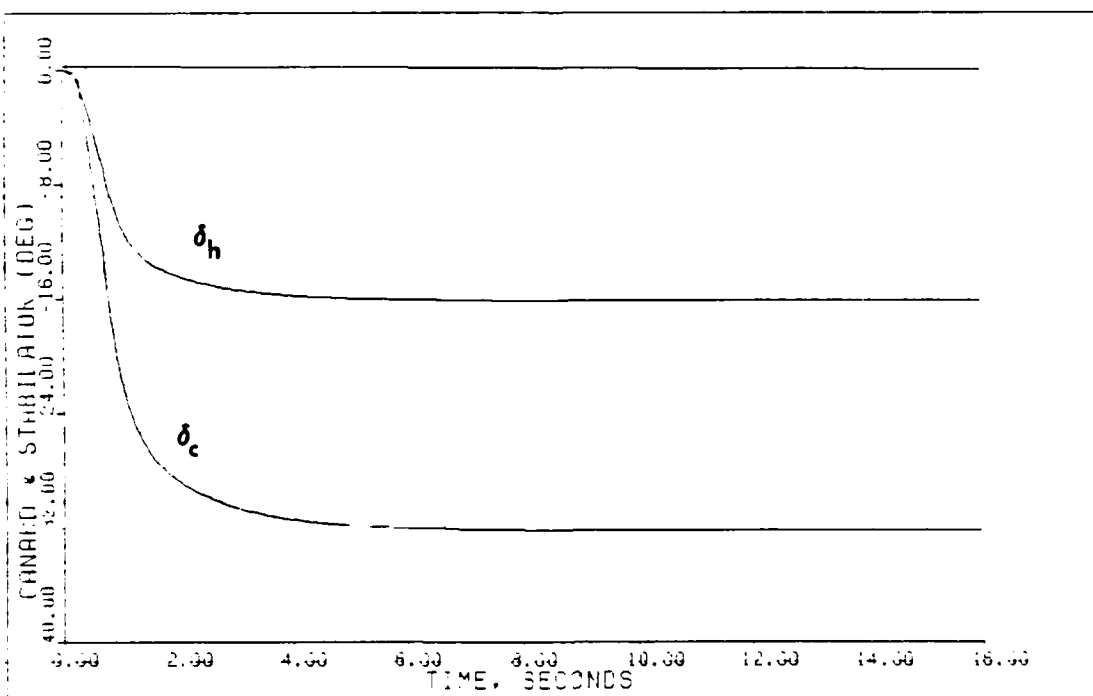
Notes:

1. See Table 5.5 for the command vector.
2. The final value of all outputs equals the commanded step input (integral control).
3. The settling times for velocity and flight path angle are estimated from the response plots.
4. Units for all angle outputs are in degrees, time is in seconds, and velocity is in feet/second.



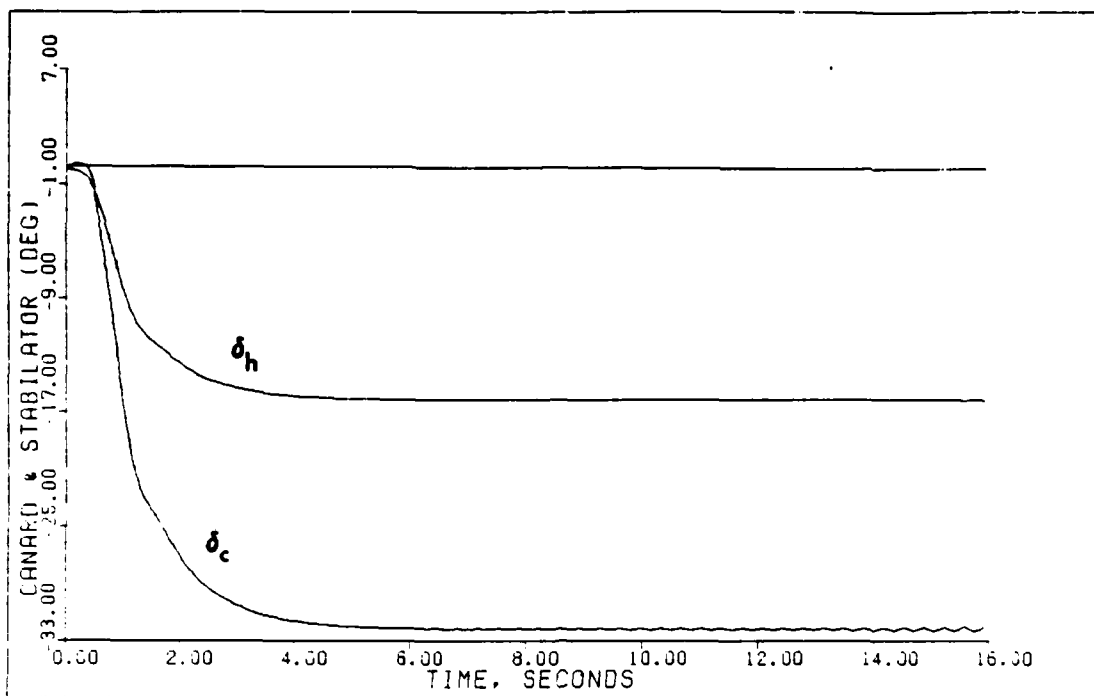
PITCH POINTING: BASIC PLANT (1.4M/FL200)

Fig. 5.37

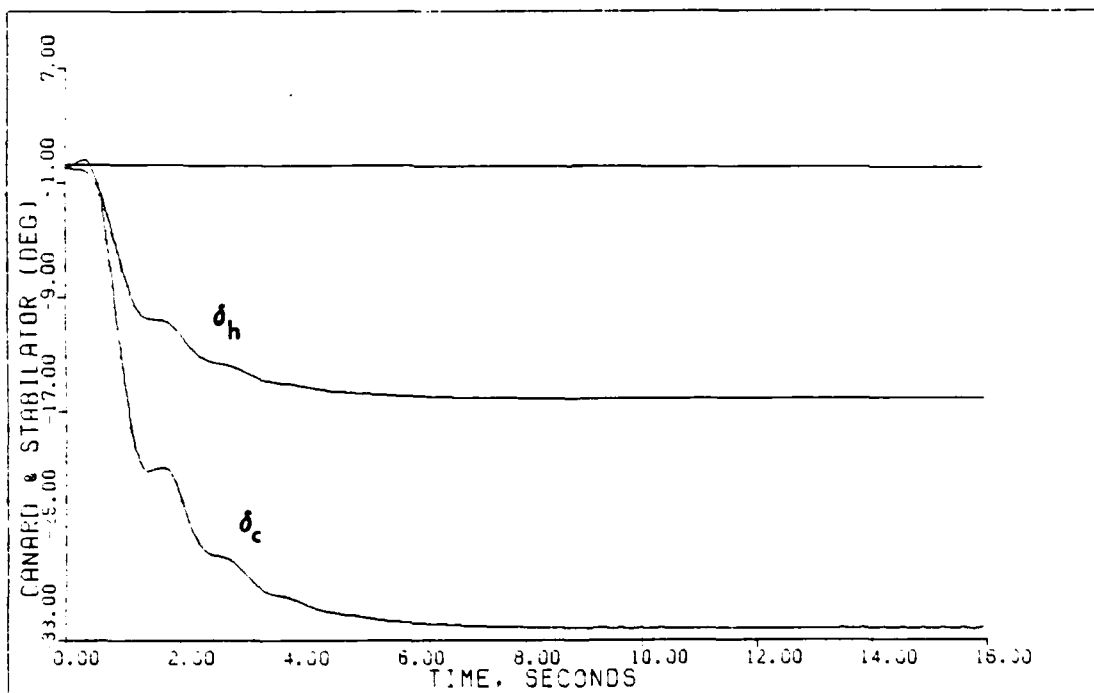


PITCH POINTING: PLANT+ACTUATORS (1.4M/FL200)

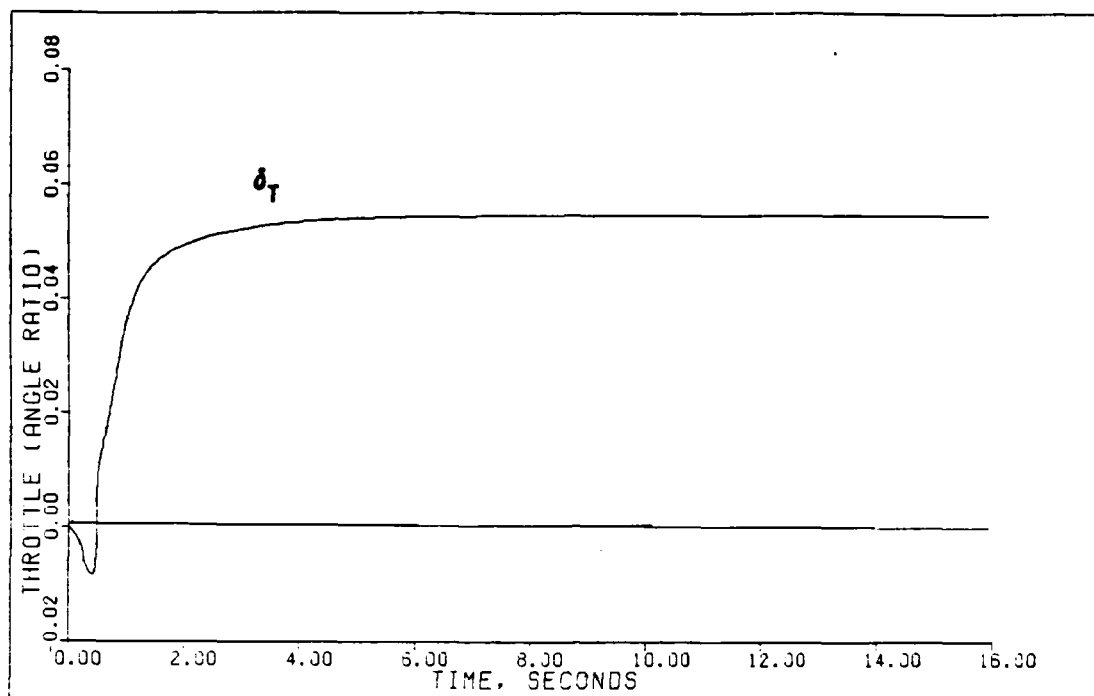
Fig. 5.38



PITCH POINTING: PLANT+ACTUATORS+DELAY (1.4M/FL200) Fig. 5.39

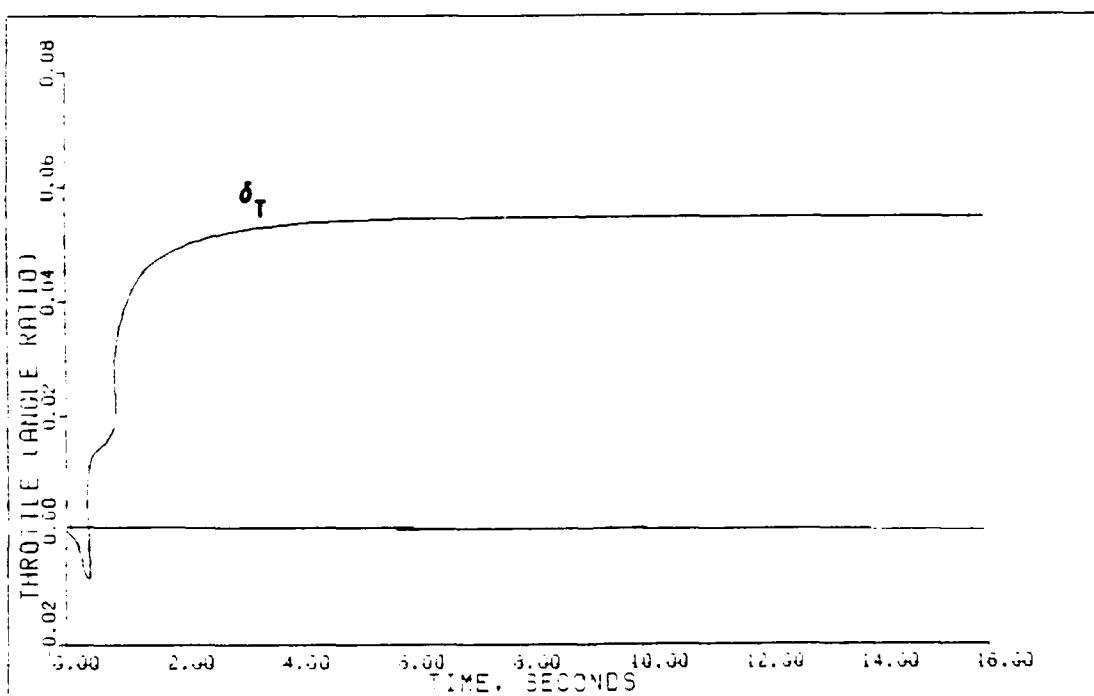


PITCH POINTING: PLANT+ACTUATORS+DELAY+SENSORS (1.4M/FL200) Fig. 5.40



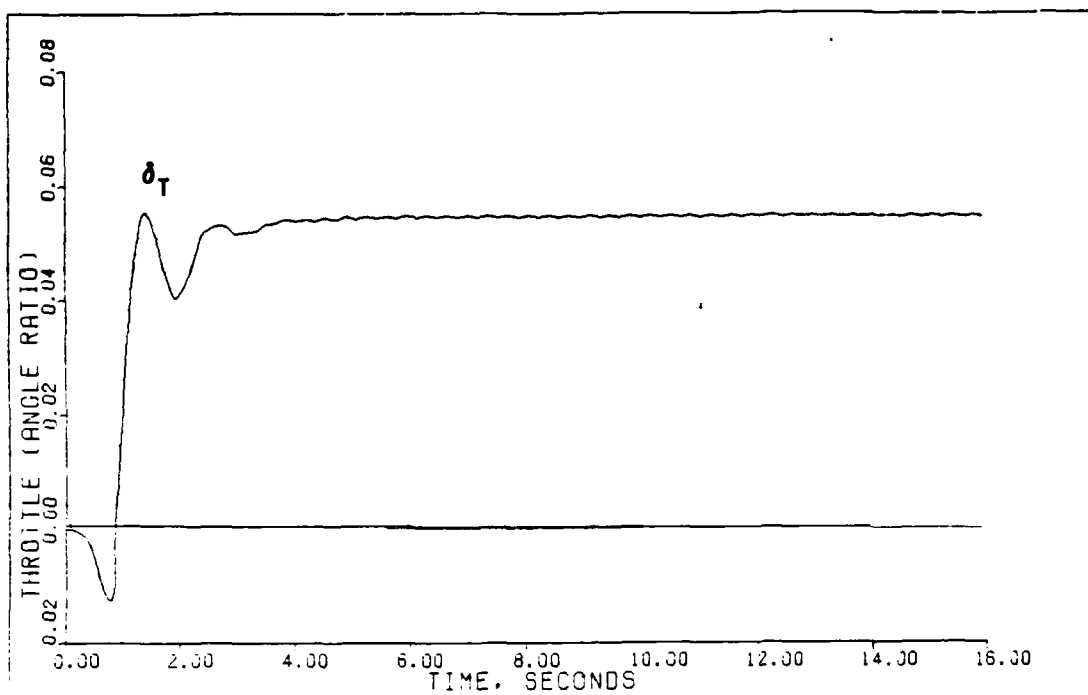
PITCH POINTING: BASIC PLANT (1.4M/FL200)

Fig. 5.41

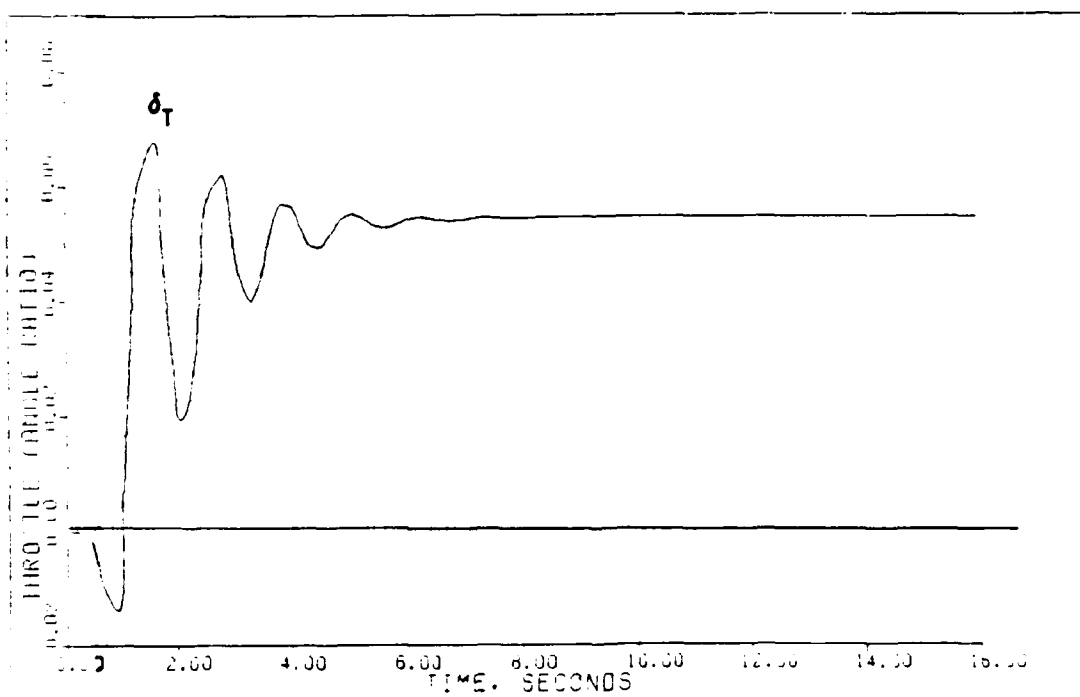


PITCH POINTING: PLANT-ACTUATORS (1.4M/FL200)

Fig. 5.42

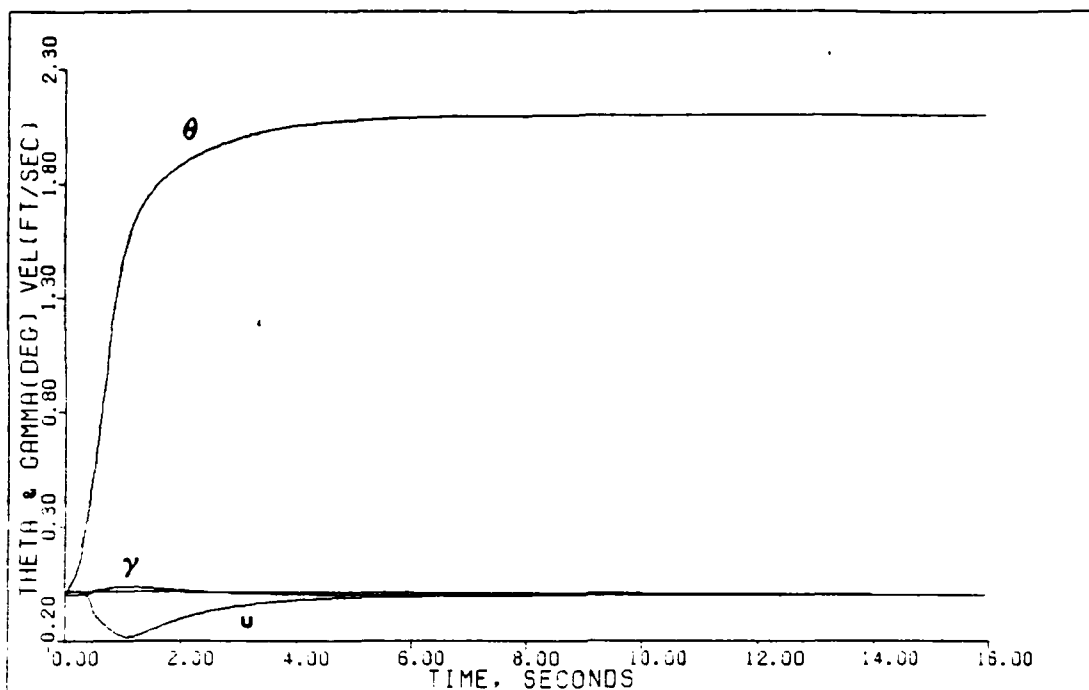


PITCH POINTING: PLANT-ACTUATORS-DELAY (1.4M/FL200) Fig. 5.43



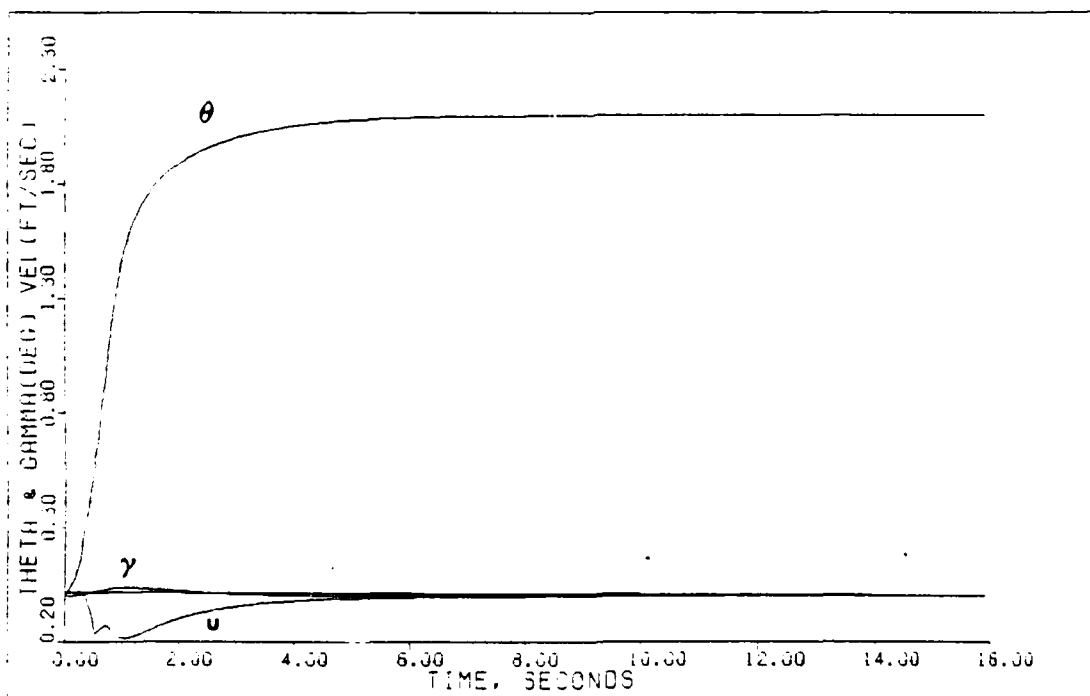
PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (1.4M, FL200)

Fig. 5.44



PITCH POINTING: BASIC PLANT (1.4M/FL200)

Fig. 5.45



PITCH POINTING: PLANT+ACTUATORS (1.4M/FL200)

Fig. 5.46

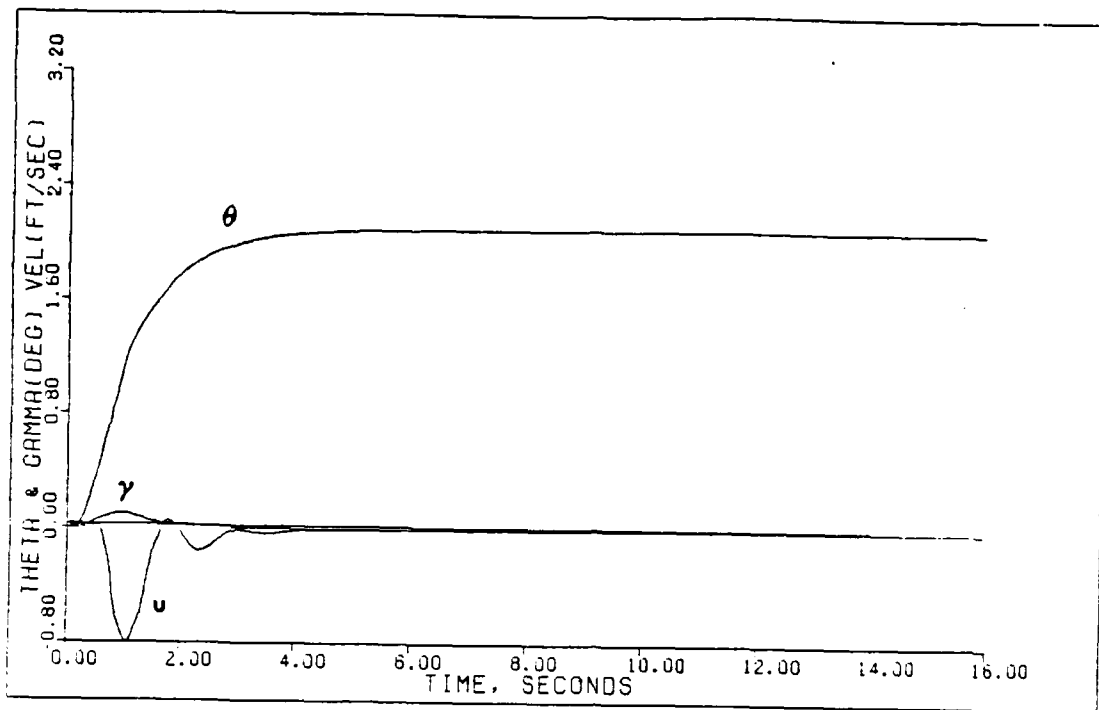


Fig. 5.47

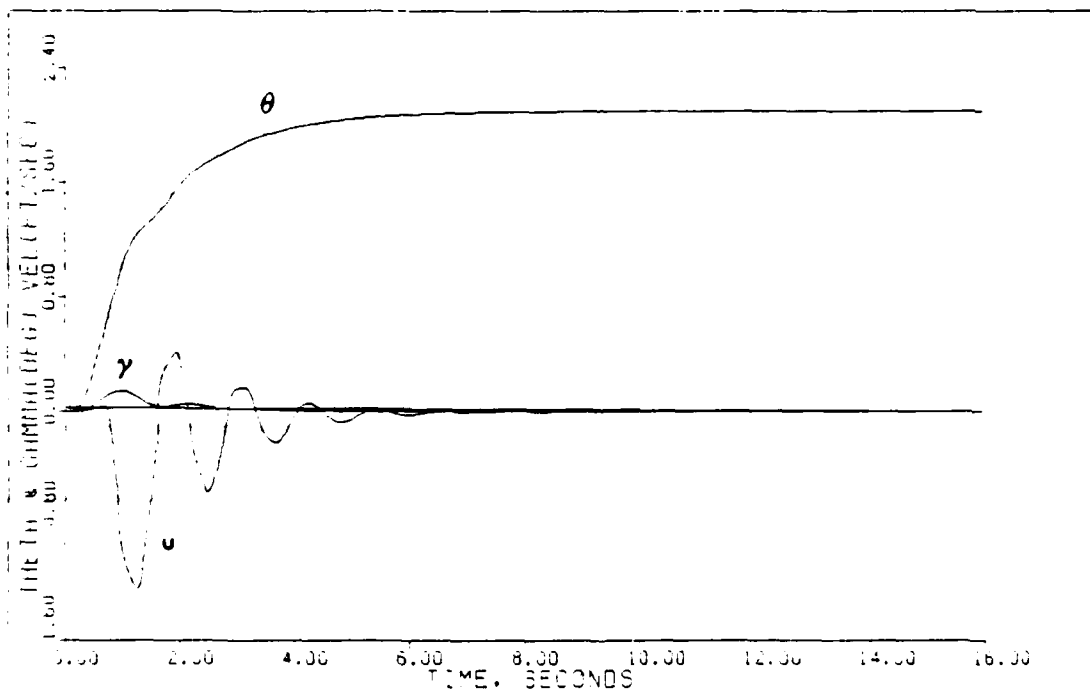
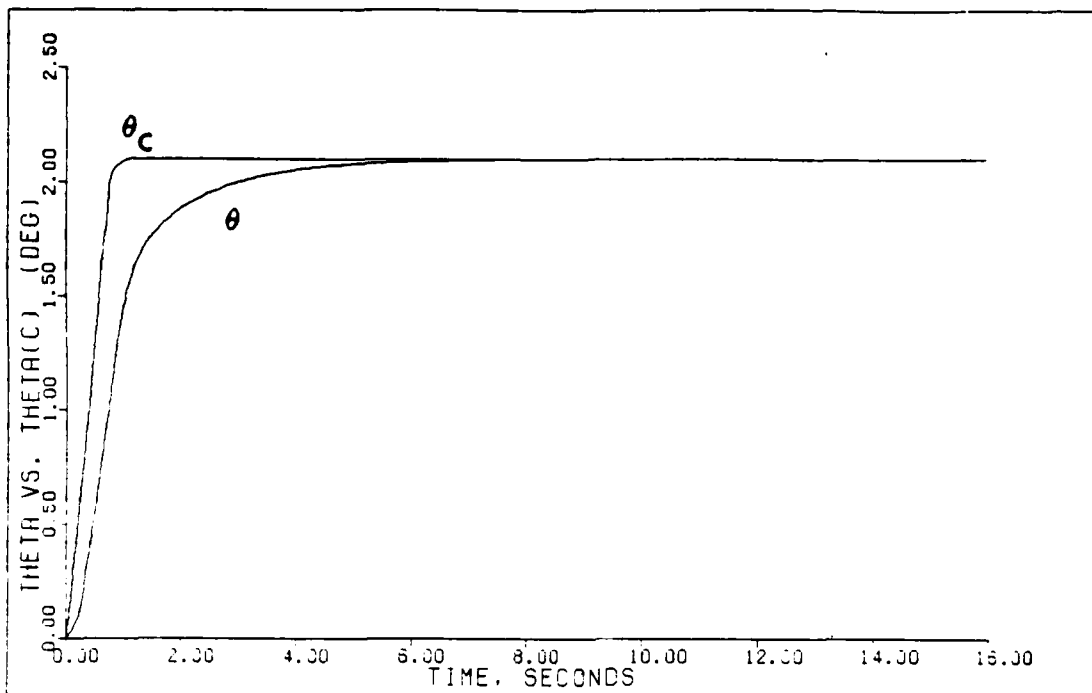
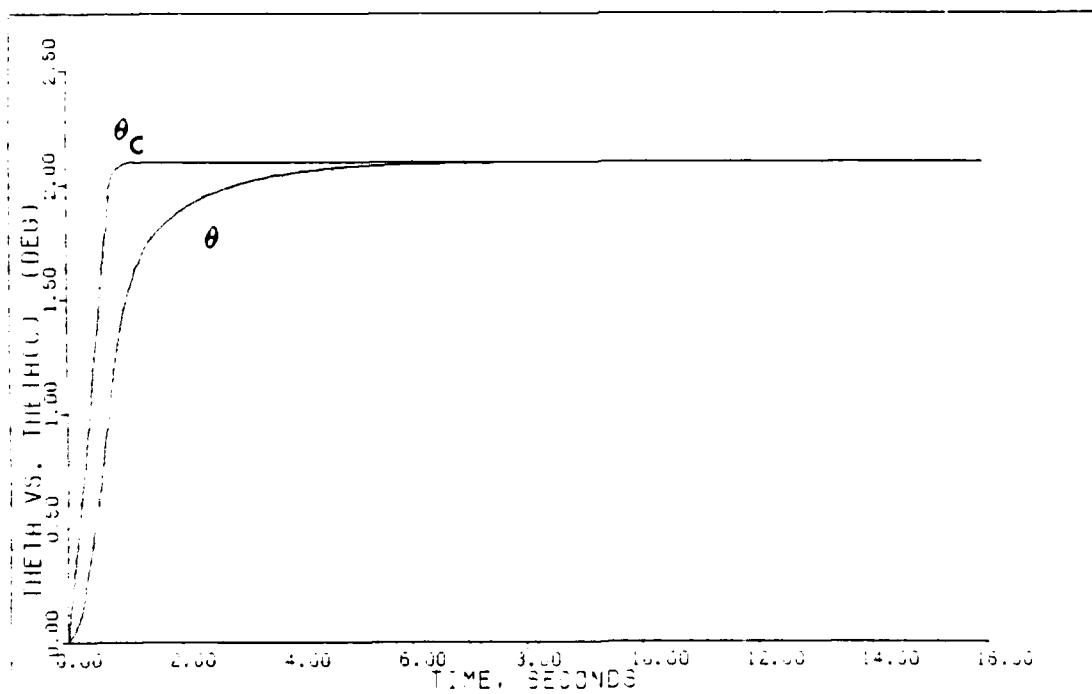


Fig. 5.48



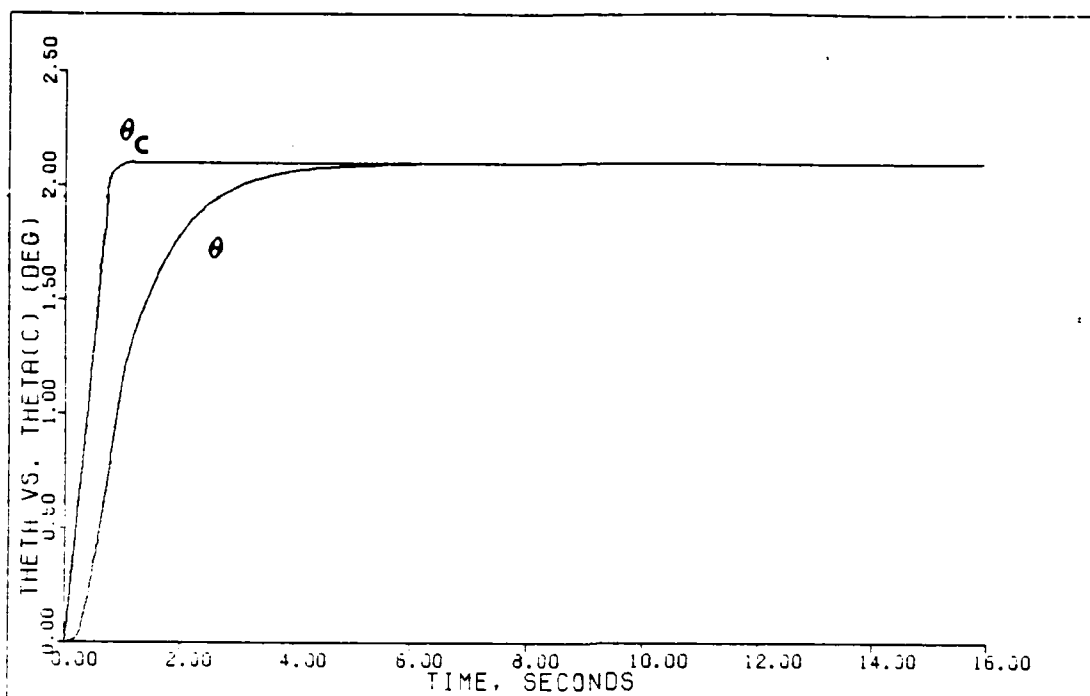
PITCH POINTING: BASIC PLANT (1.4M/FL200)

Fig. 5.49

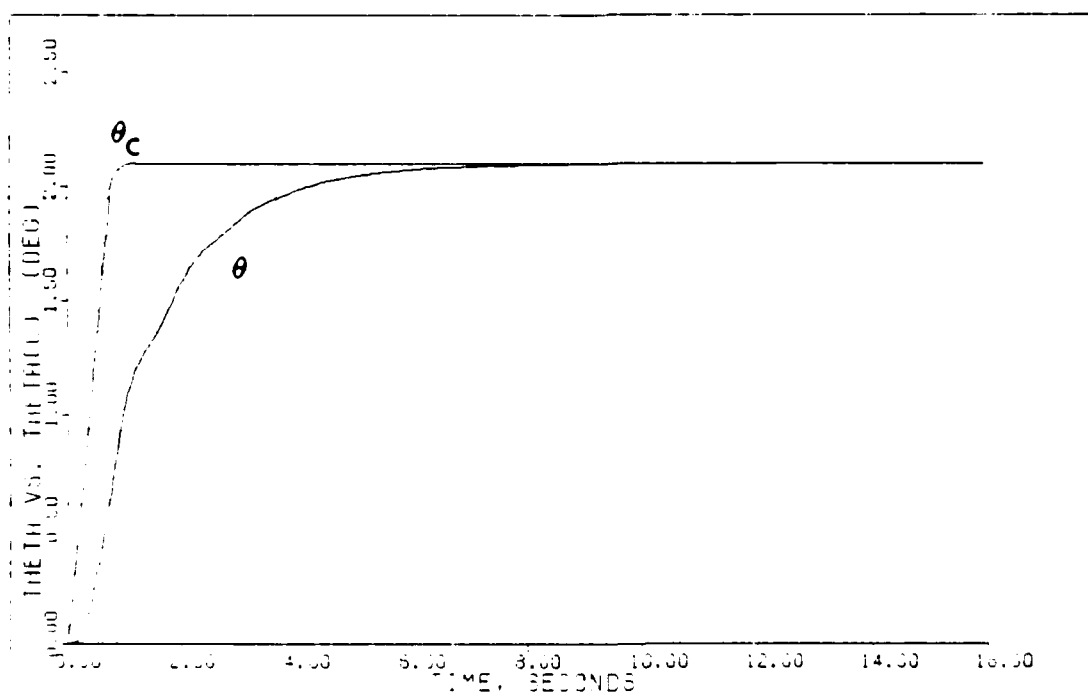


PITCH POINTING: PLANT+ACTUATORS (1.4M/FL200)

Fig. 5.50



PITCH POINTING: PLANT-ACTUATORS+DELAY (1.4M/FL200) Fig. 5.51



PITCH POINTING: PLANT-ACTUATORS+DELAY+SENSORS (1.4M/FL200)

Fig. 5.52

developed model. The variation in flight path remains insignificant (0.14 degs) and settles out within 2 secs.

Table 5.5 includes the design parameters at each stage of the controller development. Table 5.6 lists the significant figures of merit for the output responses at each point in the design. Appendix D includes the remaining designs for the pitch pointing maneuver.

5.5 Constant g Pull-Up (2.0 Mach/FL 400)

The constant g pull-up is a maneuver frequently used in air combat in a variety of ways; however, it is rarely limited to the x-z plane as is done in this study. As pointed out in Chapter III, this restriction is necessary to reduce the complexity of the analysis and provide for a straightforward simulation. It does, however, demonstrate the control law's capability at maintaining the commanded pitch rate over a simulation period short enough to prevent invalidating the linear model.

In air combat, pilots generally command a desired pitch rate using the g forces they sense as a feedback mechanism for input control. In Chapter III, Equation (3-23) relates this "pilot sensed" cockpit g force to the angular rates of the aircraft. The equation is repeated here as:

$$A_{n_p} = \dot{\gamma}U + \dot{q}\ell_x \quad (5-1)$$

In the steady-state (i.e. constant pitch rate and AOA), this equation reduces to:

$$A_{n_p} = [U/1845]q \quad (5-2)$$

which relates cockpit acceleration to pitch rate by a simple proportionality constant. By ramping the input theta command at a slope equal to the appropriate pitch rate, a constant steady-state g force is commanded.

This scheme is used to indirectly command a desired g force through q without having to include acceleration as a state in the aircraft model or in the commanded output. Since the aircraft is already at 1.0 g in equilibrium flight, a bias of 1 g is added to the computation of cockpit g's for plotting purposes. Consequently, for a 2 g command, the ramp slope for theta is equal to 1 g, with the simulation beginning at an initial value of 1 g.

The constant g pull-up is simulated for both a 2 g and 9 g command input. The 2 g simulations demonstrate the controller's capabilities throughout the full range of model complexity. The 9 g simulation is presented as a demonstration of the aircraft's control surface capabilities, recognizing that the linear model is only valid for the first few seconds of these simulations. With this intent in mind, the 9 g simulation is performed using only the basic aircraft model. Tables 5.7 and 5.8 list the controller design parameters and the response figures of

merit, respectively, for the 2 g command. Tables 5.9 and 5.10 list the same data for the 9 g command.

2.0 g Command Input. The control inputs for the 2 g pull-up using the basic model are shown in Figure 5.53. As seen, neither control input limits the magnitude of this maneuver. The canard smoothly deflects in the positive direction (after a minor negative transient) to begin the climbing maneuver. The combined stabilator-nozzle input also deflects downward (trailing edge) but is now a more powerful control input than the original stabilator in the four-state model. As a result, it balances the moment and forces caused by the canard with a much smaller steady-state value.

The output theta, follows the commanded ramp input with a constant error of 0.31 degs. This result is expected since the system is Type 1 (single integrator in the forward loop). A Type 1 system can track only a step input with zero steady-state error. The resulting error in theta is inversely proportional to the forward loop gain. This error is not significant, however, since the pilot commands a rate (\dot{q}) and not a specific pitch attitude.

The output pitch rate, however, is of primary importance to the pilot since he sees this by the movement of the aircraft's nose. Generally, any oscillations in pitch rate or g loading can result in PIO's (pilot-induced

oscillations) as he attempts to control the variations.

Figure 5.61 shows a rapid rise (0.95 secs) in q with negligible overshoot and a quick settling time (2.5 secs). This is an ideal response in a fighter aircraft; however, the model lacks the additional dynamics present in the real aircraft.

The change in flight path angle quickly reaches a constant value of approximately 1 deg/sec (Figure 5.69). From Equation (5-1), cockpit acceleration should also be constant once the change in pitch rate goes to zero. Figure 5.73 plots the cockpit acceleration over time with very satisfactory results. The small overshoot along with the rapid settling time would not present any control problems to the pilot.

Once again, the addition of actuator dynamics to the three-state model does not degrade the overall system response. In fact, the responses are indistinguishable from those of the basic aircraft (Figures 5.54, 5.58, 5.62, 5.66, 5.70, and 5.74).

To add to the complexity of the model, computational time delay is included in the simulation. This increased delay has a significant affect on the response time of the control inputs (Figure 5.55). The overall gain must be reduced ($\epsilon = 0.1998$) to maintain stability which slows down the response of the entire system. Likewise, the integral gain is also reduced from $(2.0)K_0$ to

$(1.11)K_0$ which increases the settling time of all of the outputs and states. As a result, θ is slightly slower in tracking its commanded ramp input (Figure 5.59) while the significant overshoot in α (Figure 5.67) causes a minor variation in flight path rate (Figure 5.71).

All of these variations affect the desired cockpit g loading (Figure 5.75) but none of them is as critical as the influence of pitch rate. The acceleration of the aircraft's center of mass is dependent only on the rate-of-change of the flight path angle (assuming constant velocity). Cockpit acceleration, however, is dependent on not only the acceleration of the CG but also on additional accelerations caused by the change in pitch rate multiplied by the distance the pilot sits from the aircraft CG (Equation 5-1). Figure 5.71 shows a fairly constant change in γ (acceleration of the CG), however, the pitch rate is not constant (Figure 5.63). In fact, since \dot{q} never reaches zero, its effect is felt in cockpit g throughout the entire simulation (Figure 5.75). Fortunately, since the overshoot in A_{n_p} is small with a low oscillation frequency, the pilot could easily control the response.

Adding sensor dynamics does not appreciably change the system response over the previous addition of computational delay. The design parameters are not modified since any increase in gain, used to reduce settling times, would result in instability.

TABLE 5.7

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Constant g Pull-Up (2.0 g's)

Flt Condition: 2.0 Mach at FL 400

Command Vector \underline{v} : v_1 = Theta: 20, 0.3316, 20, 20
 v_2 = Alpha: 1.5, 0.008727, 20, 20

Basic Plant

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
2.000	0.950	1.0	.1265E+02	-.6523E+02
		0.02	-.2880E+01	-.1451E+02

Plant + Actuators

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
2.000	0.950	1.0	.1265E+02	-.6523E+02
		0.05	-.2880E+01	-.1451E+02

Plant + Actuators + Delay

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
1.111	0.1998	1.0	.2660E+01	-.8231E+01
		0.03	-.6057E+00	-.1831E+01

Plant + Actuators + Delay + Sensors

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
1.111	0.1998	1.0	.2660E+01	-.8231E+01
		0.03	-.6057E+00	-.1831E+01

Notes:

- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\alpha)\underline{K}_0$.
- Irregular Design: $\underline{M} = (0.3, 0, 0)^T$

TABLE 5.8
DESIGN OUTPUT FIGURES OF MERIT

Maneuver: Constant g Pull-Up (2.0 g)

Flt Condition: 2.0 Mach at FL 400

Basic Plant

<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Acceleration	+2.15	2.4	3.8
Angle of Attack	+0.579	1.92	3.15
Pitch Rate	+0.95	2.0	2.5

Plant + Actuators

<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Acceleration	+2.1	2.4	3.5
Angle of Attack	+0.58	1.92	3.15
Pitch Rate	+0.95	2.0	2.2

Plant + Actuators + Delay

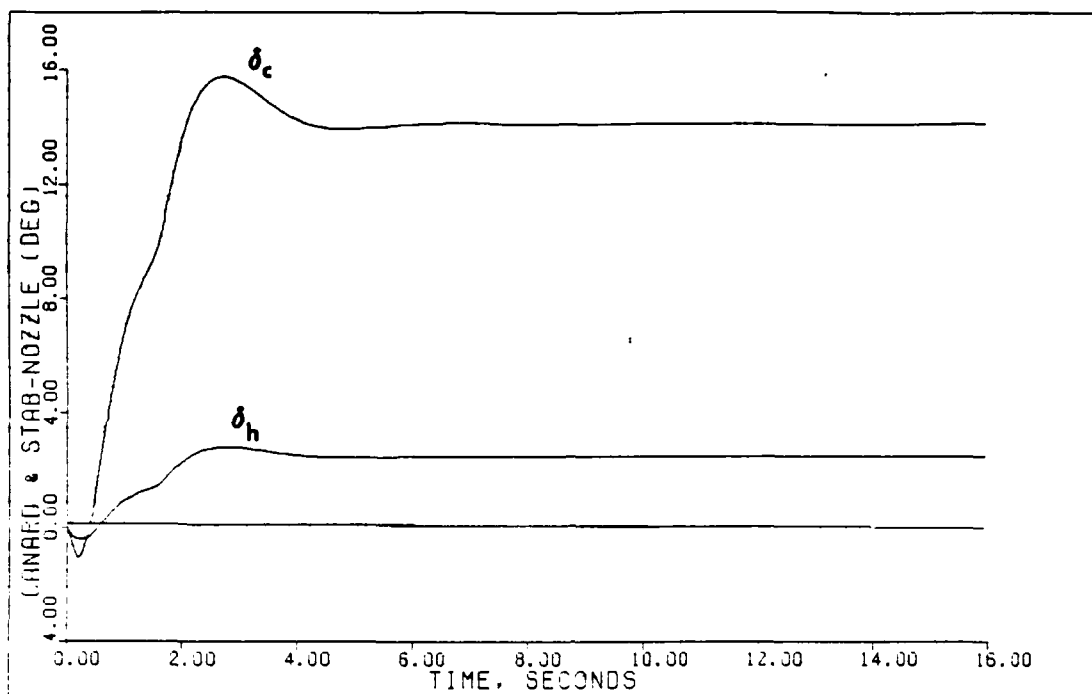
<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Acceleration	+2.3	5.8	10.5
Angle of Attack	+1.12	5.07	14.7
Pitch Rate	+1.2	4.6	9.0

Plant + Actuators + Delay + Sensors

<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Acceleration	+2.3	5.9	9.8
Angle of Attack	+1.15	4.9	14.87
Pitch Rate	+1.2	4.6	9.5

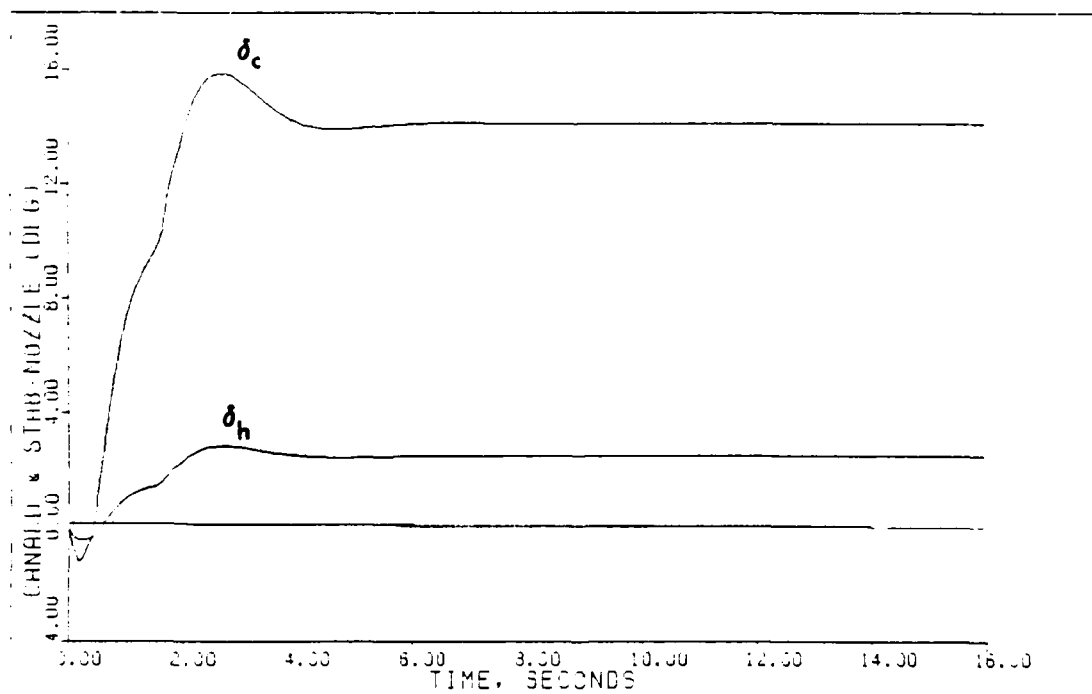
Notes :

1. See Table 5.7 for the command vector.
2. The final value of angle of attack equals the commanded step input (integral control).
3. All values for acceleration and pitch rate are estimated from the response plots.
4. Units for alpha are in degrees, time in seconds, pitch rate in degrees/second, and acceleration in g's.



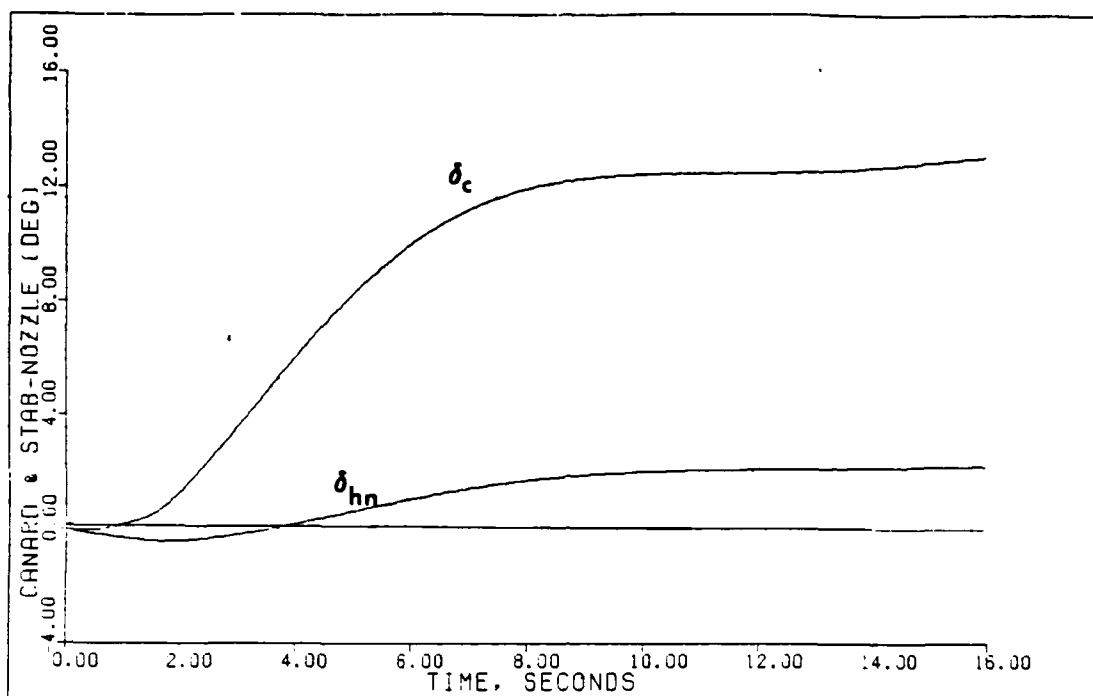
2G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.53



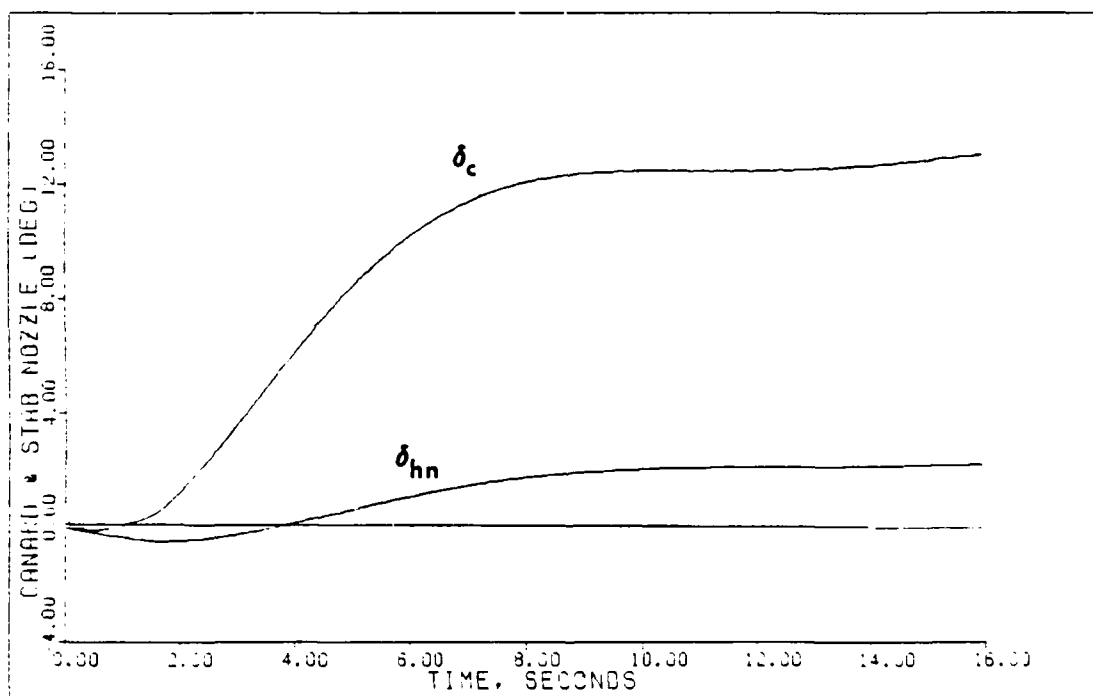
2G PULL-UP: PLANT+ACTUATORS (2.0M/FL400)

Fig. 5.54



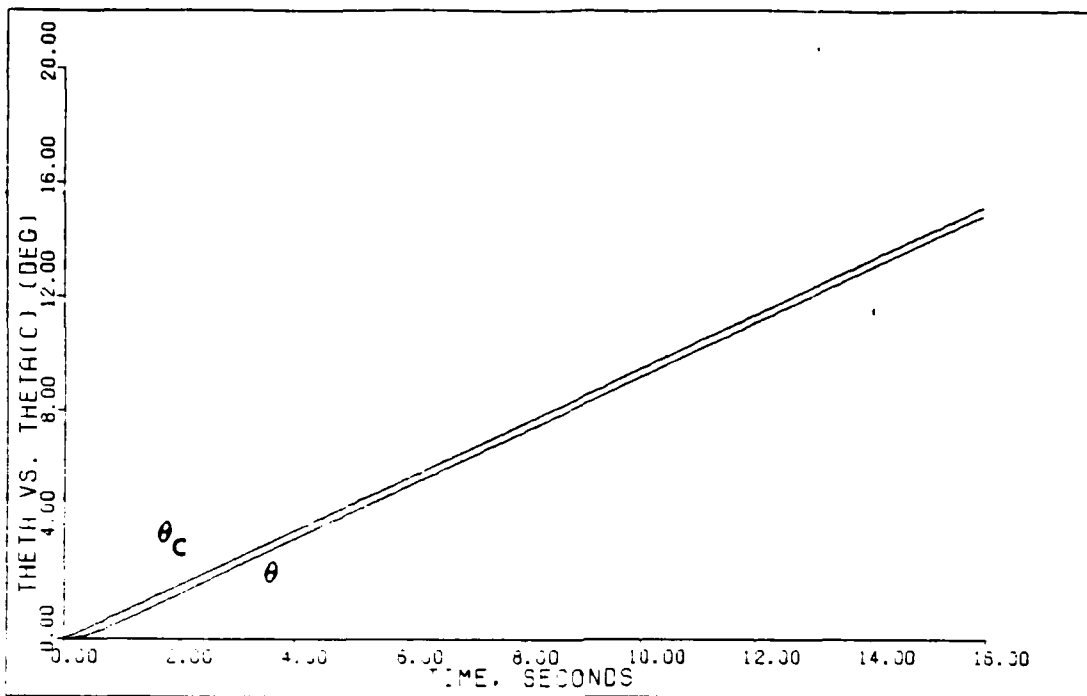
2G PULL-UP: PLANT+ACTUATORS+DELAY (2.0M/FL400)

Fig. 5.55



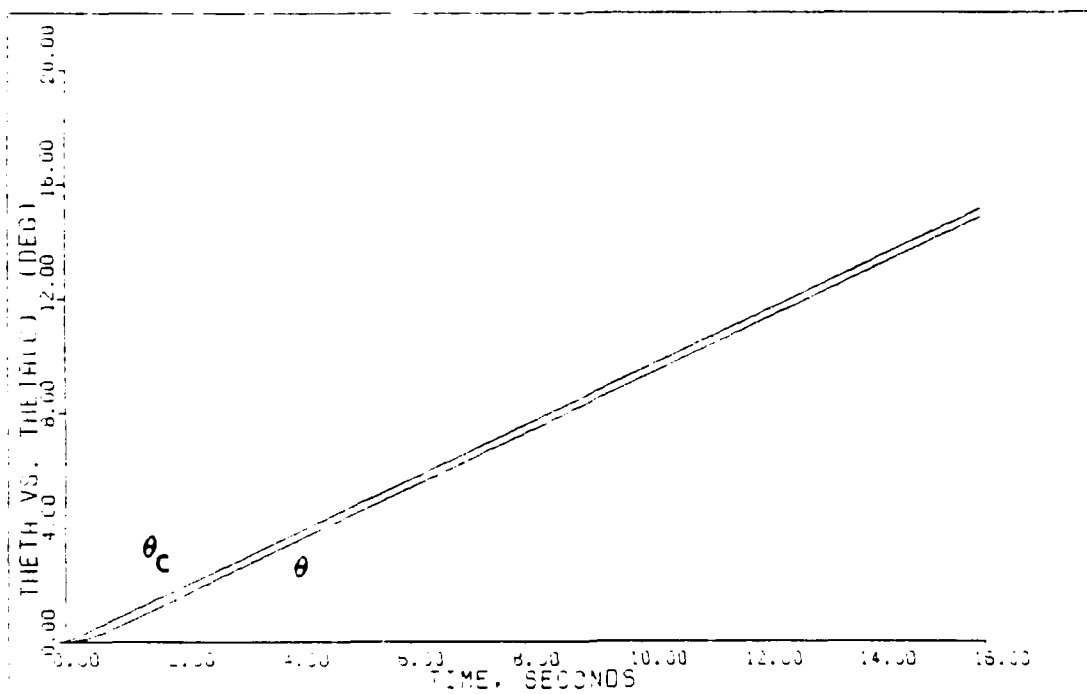
2G PULL-UP: PLANT+ACTUATORS+DELAY+SENSORS (2.0M/FL400)

Fig. 5.56



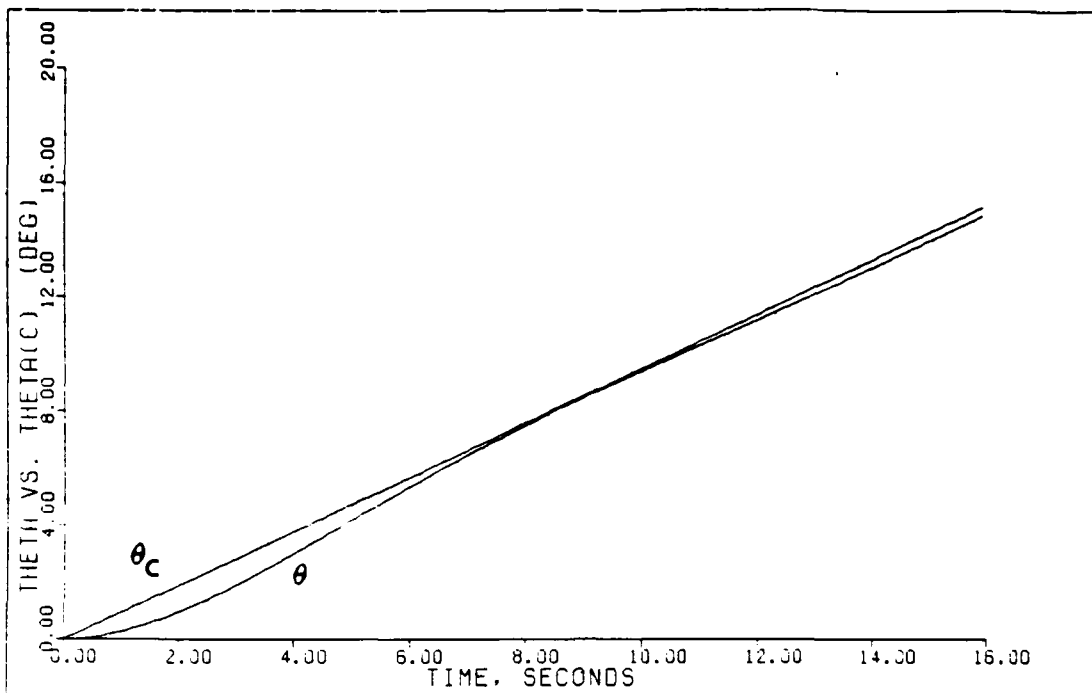
CG PULL-UP: B4310 PLANT (2.0M/FL400)

Fig. 5.57



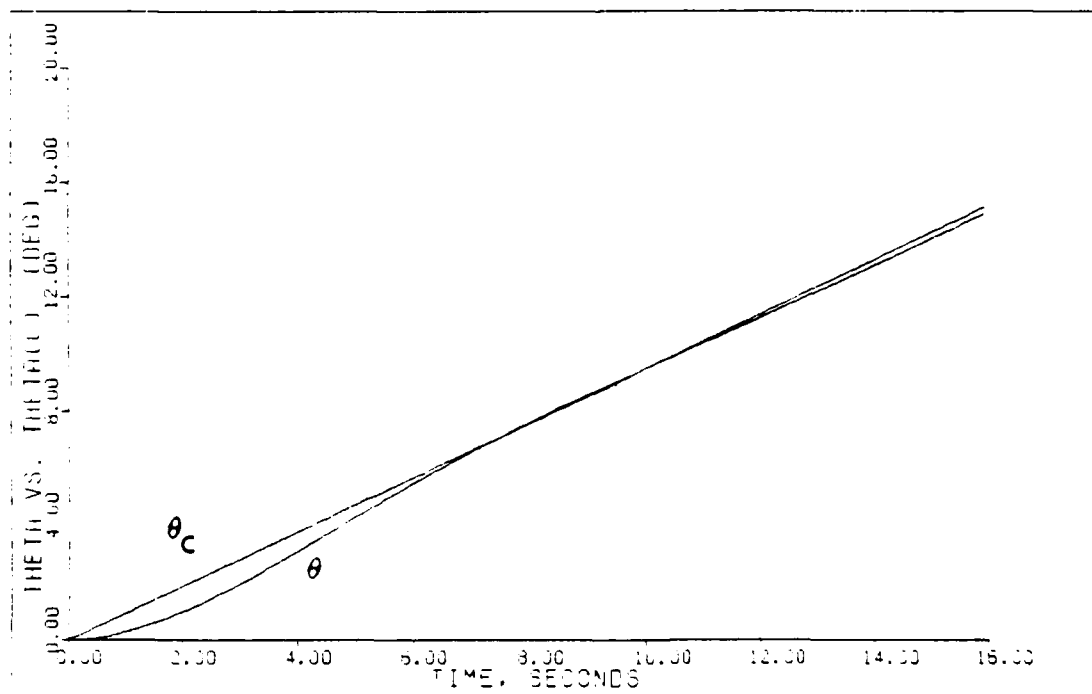
CG PULL-UP: PLANT-ACTUATORS (2.0M/FL400)

Fig. 5.58



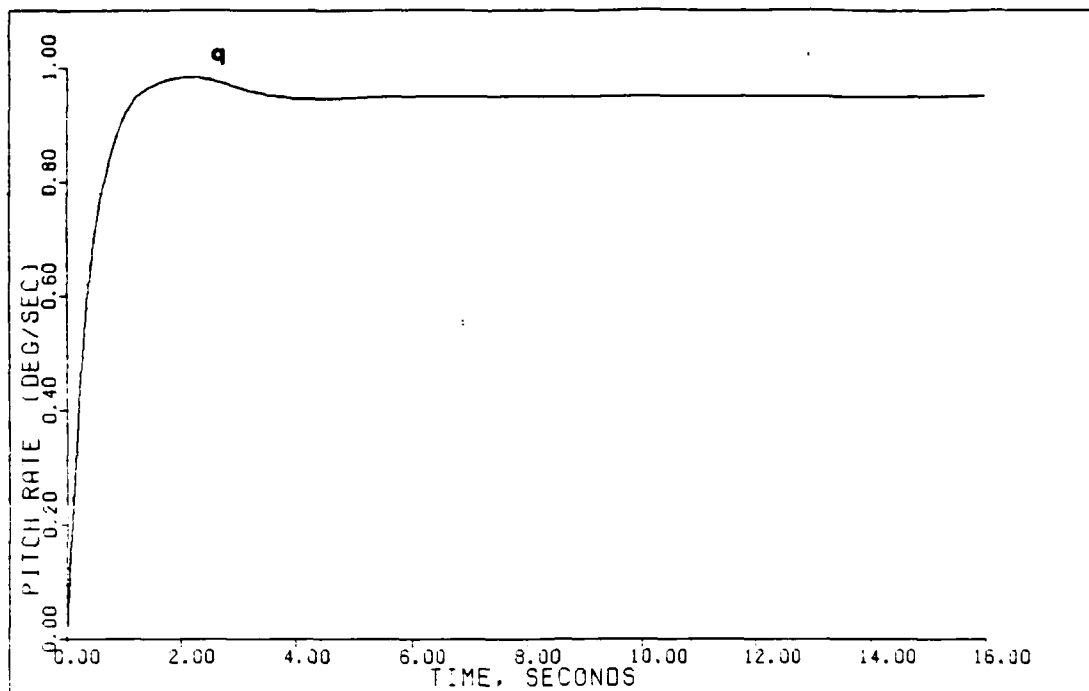
23 PULL-UP: PLANT+ACTUATORS+DELAY (2.0M/FL400)

Fig. 5.59



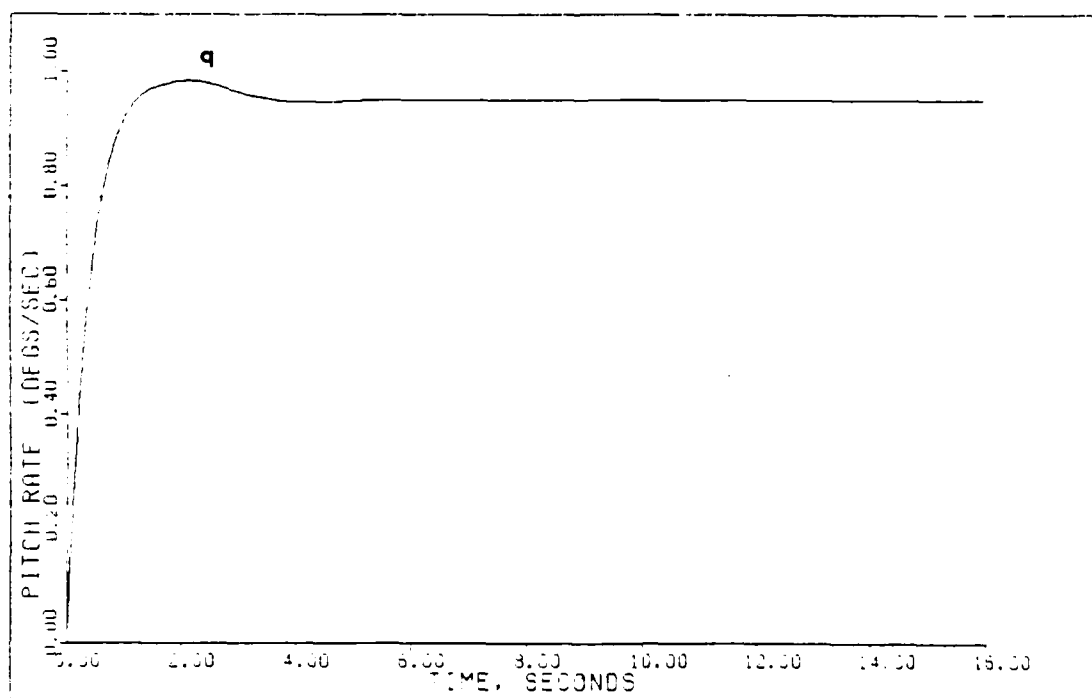
23 PULL-UP: PLANT+ACTUATORS+DELAY+SENSORS (2.0M/FL400)

Fig. 5.60



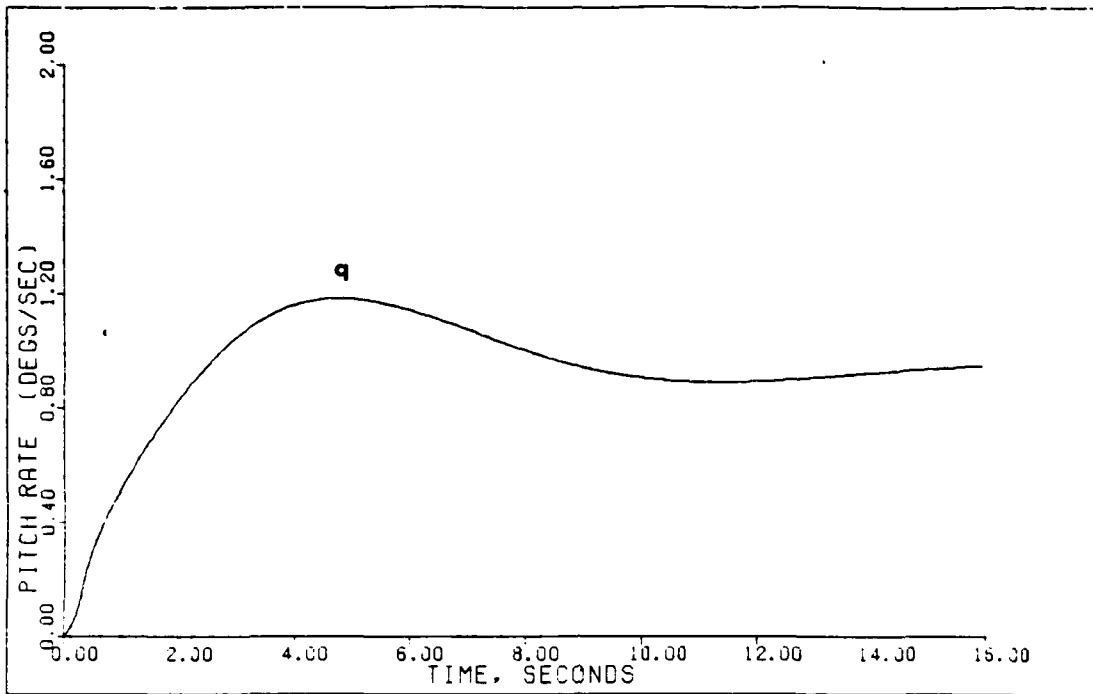
2G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.61



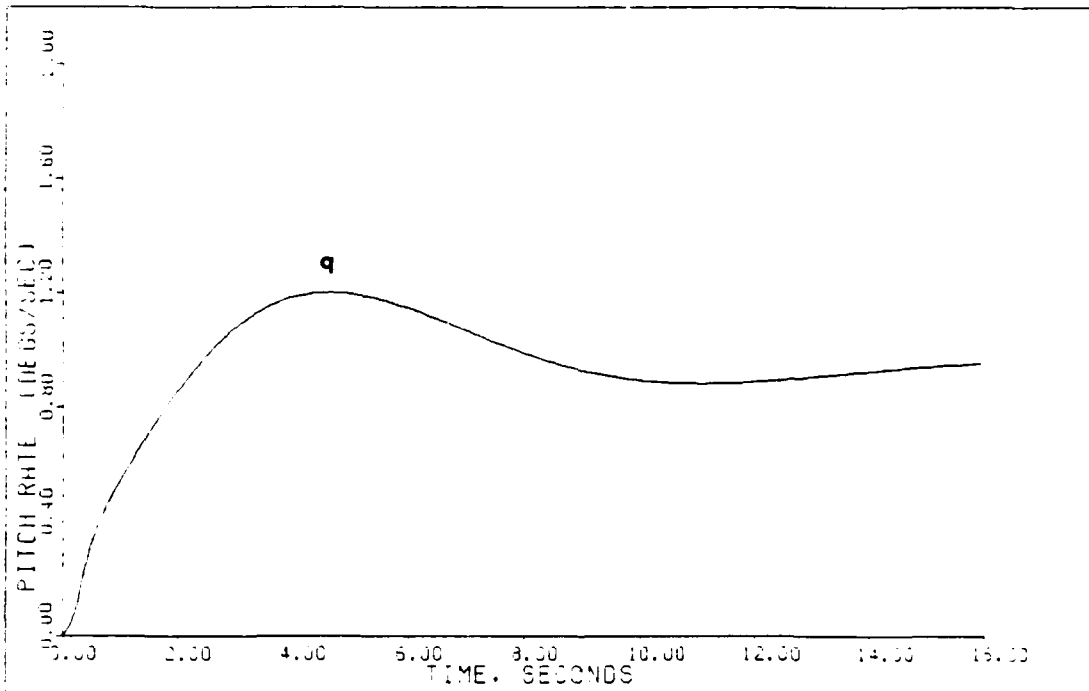
2G PULL-UP: PLANT-ACTUATORS (2.0M/FL400)

Fig. 5.62



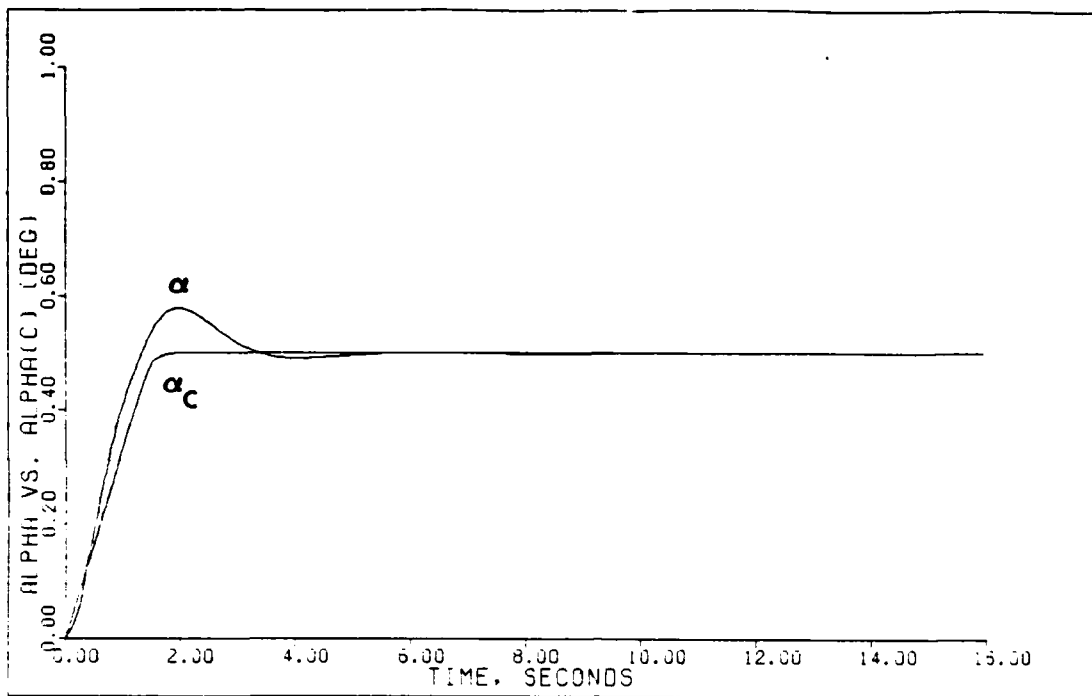
2G PULL-UP: PLANT+ACTUATORS+DELAY (2.0M/FL400)

Fig. 5.63



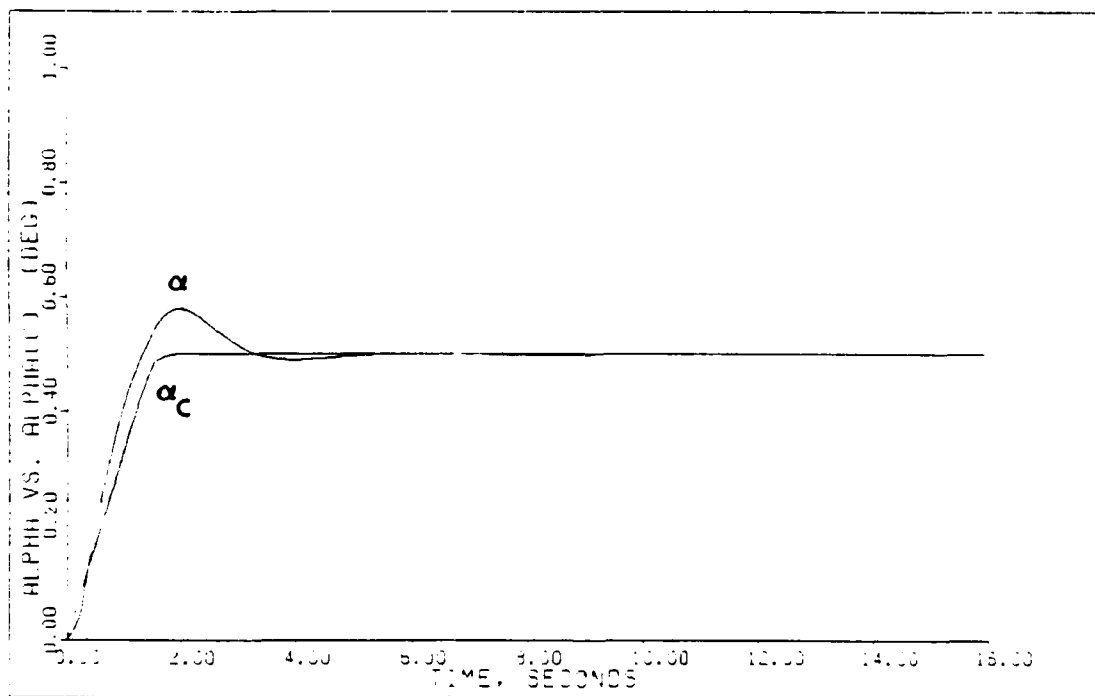
2G PULL-UP: PLANT+ACTUATORS+DELAY+SENSORS (2.0M/FL400)

Fig. 5.64



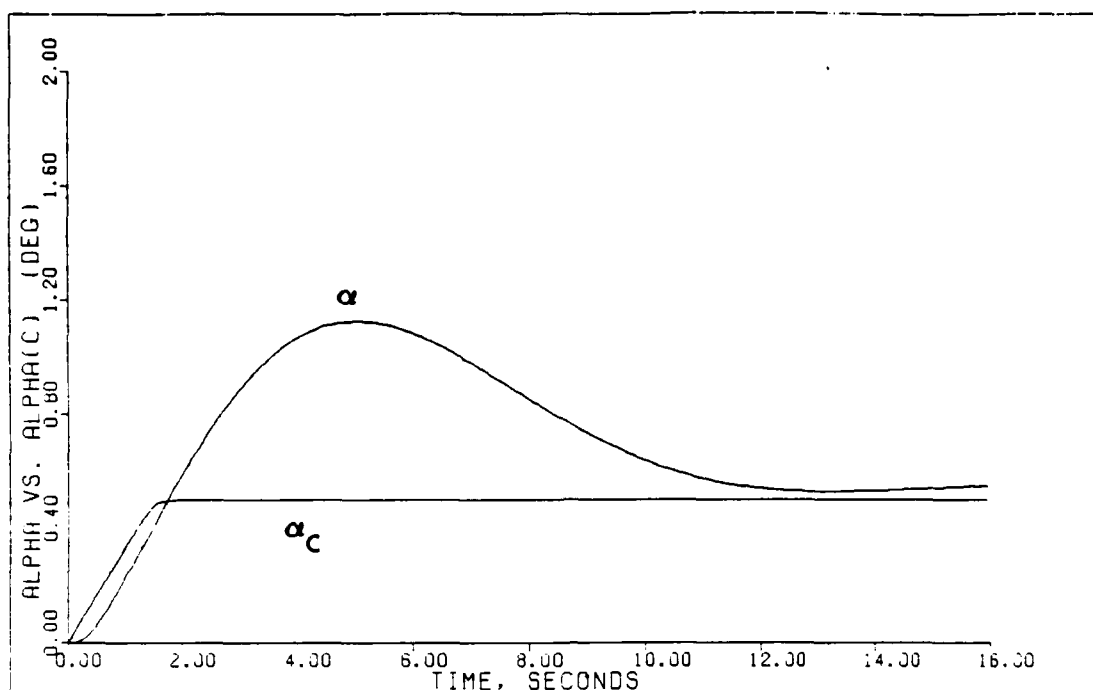
2G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.65



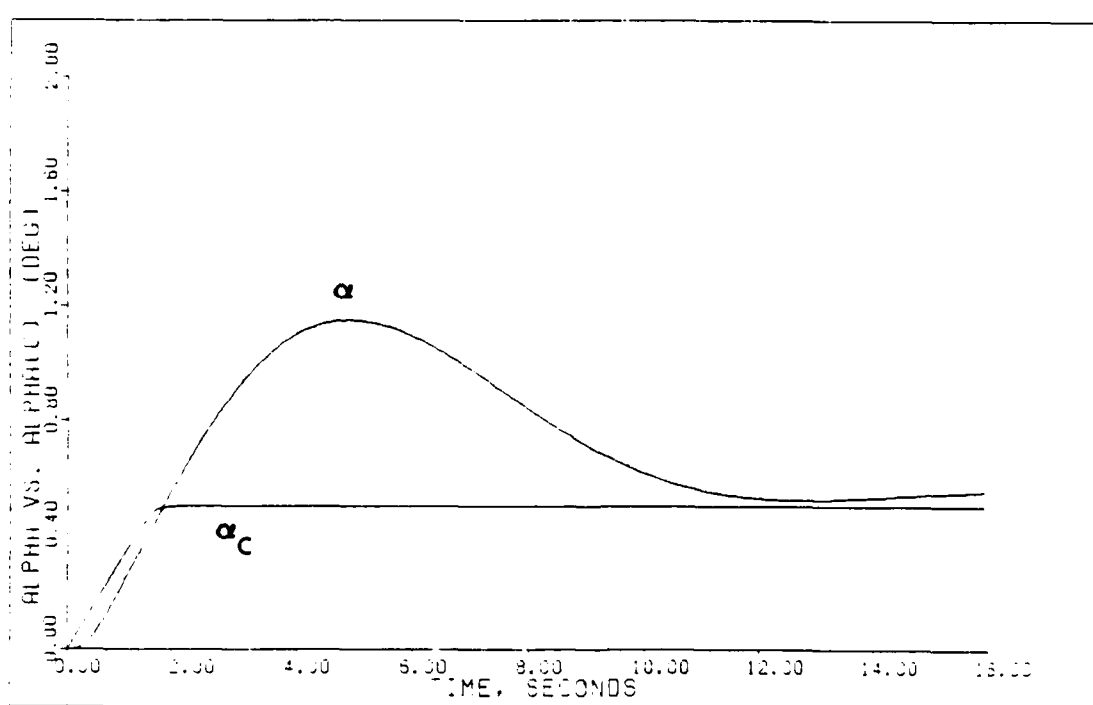
2G PULL-UP: PLANT+ACTUATORS (2.0M/FL400)

Fig. 5.66



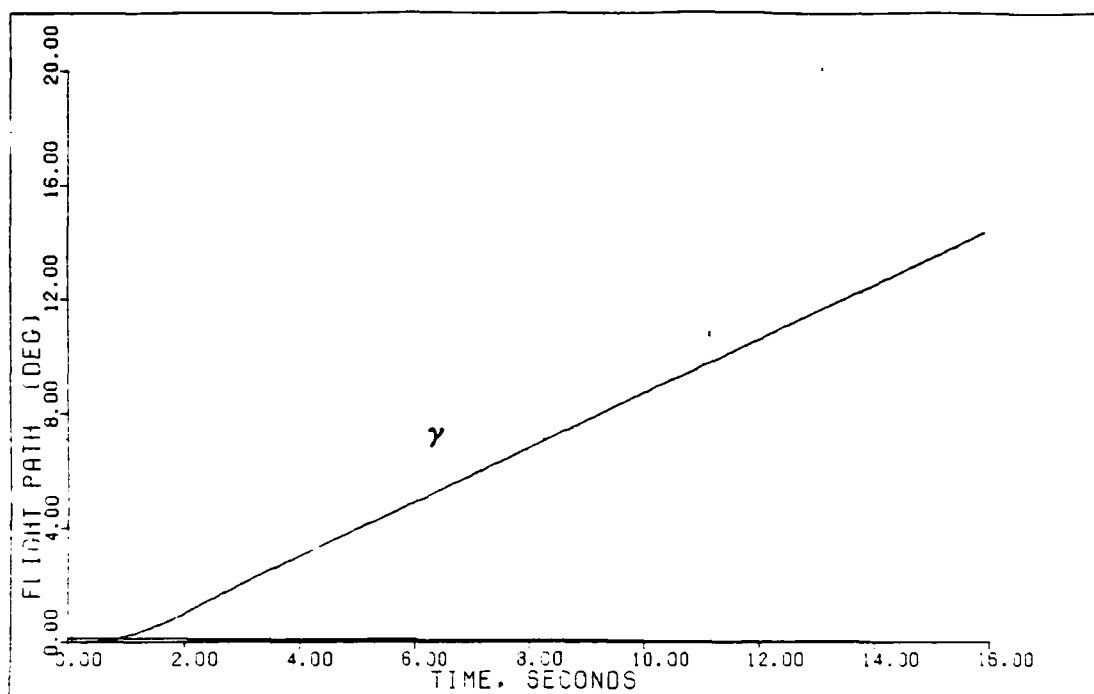
2G PULL-UP: PLANT+ACTUATORS+DELAY (2.0M/FL400)

Fig. 5.67



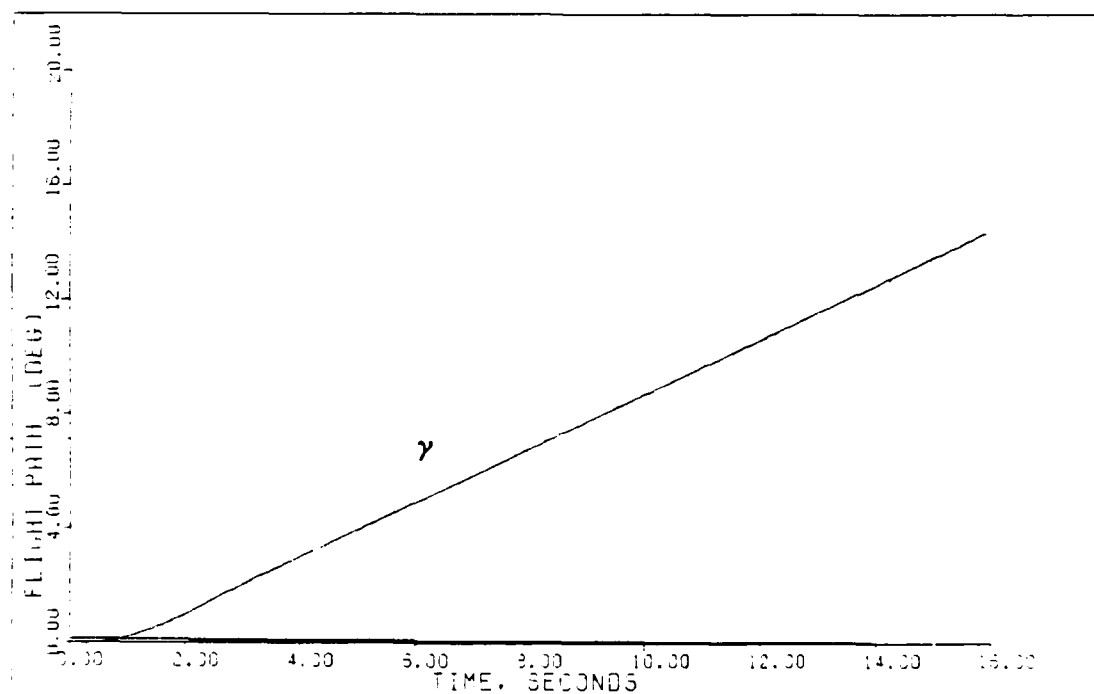
2G PULL-UP: PLANT+ACTUATORS+DELAY+SENSORS (2.0M/FL400)

Fig. 5.68



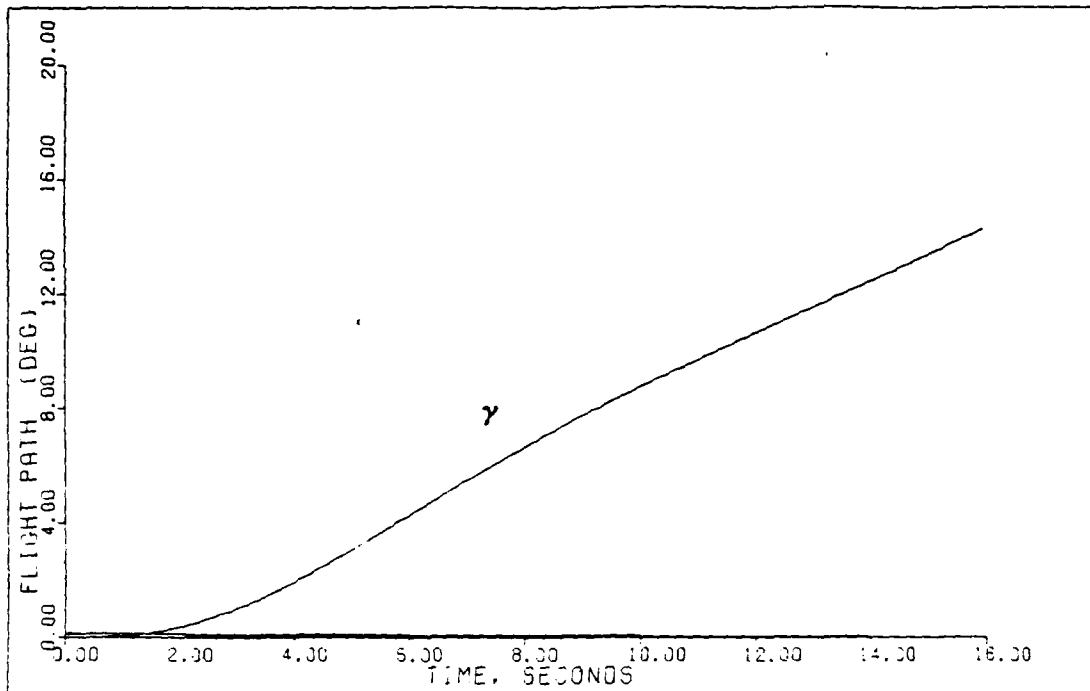
2G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.69



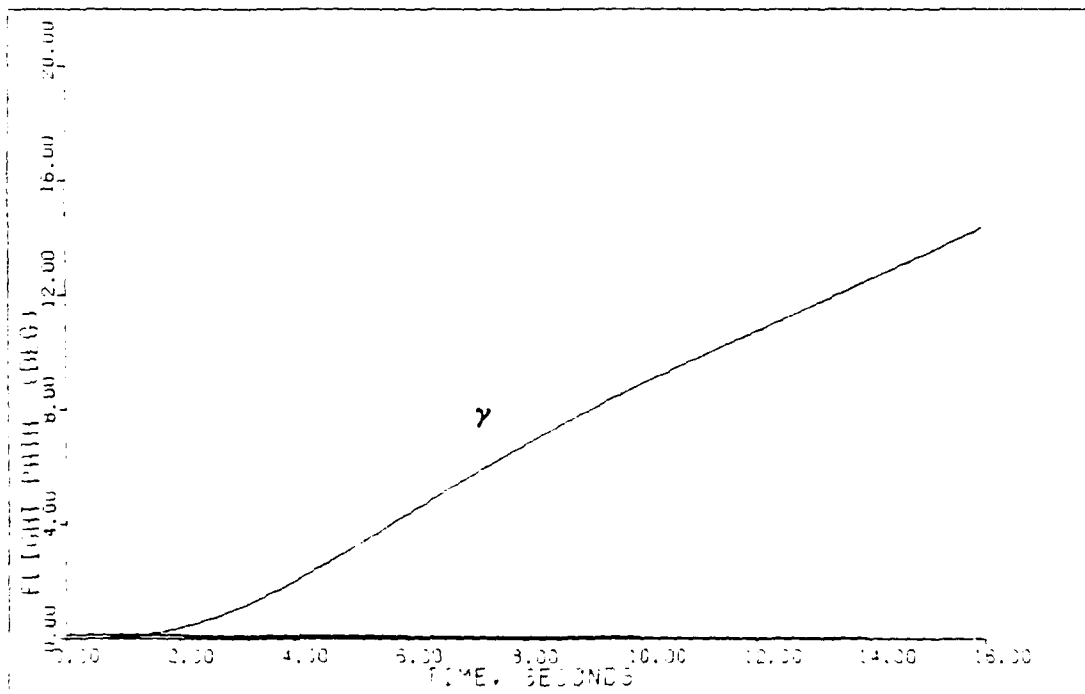
2G PULL-UP: PLANT+ACTUATORS (2.0M/FL400)

Fig. 5.70



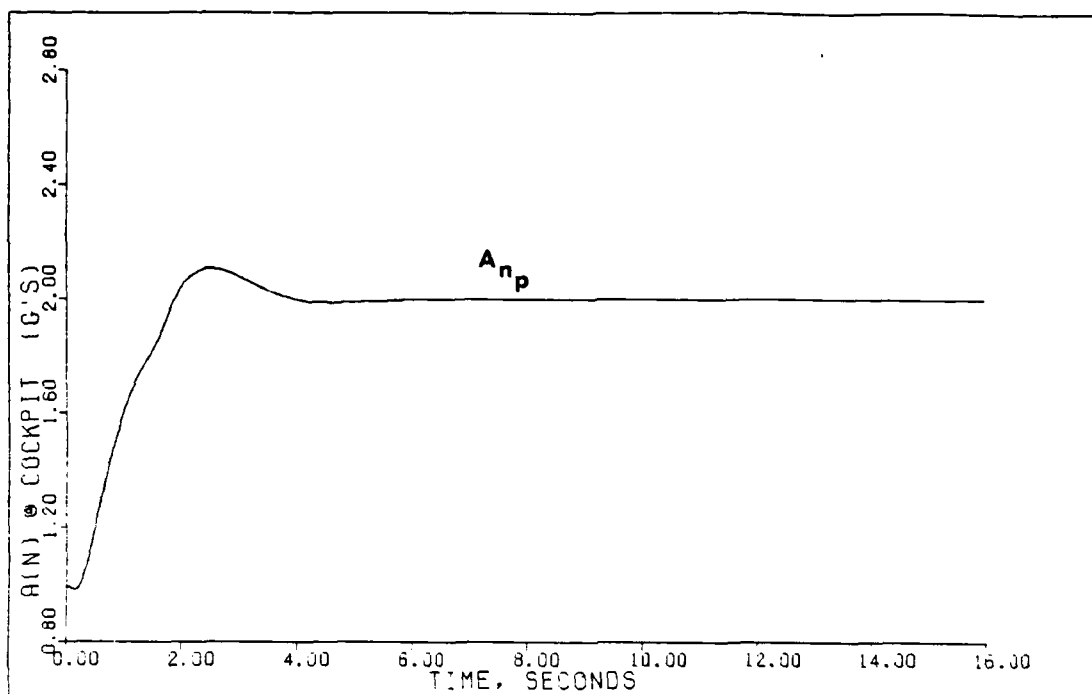
2G PULL-UP: PLANT+ACTUATORS+DELAY (2.0M/FL400)

Fig. 5.71



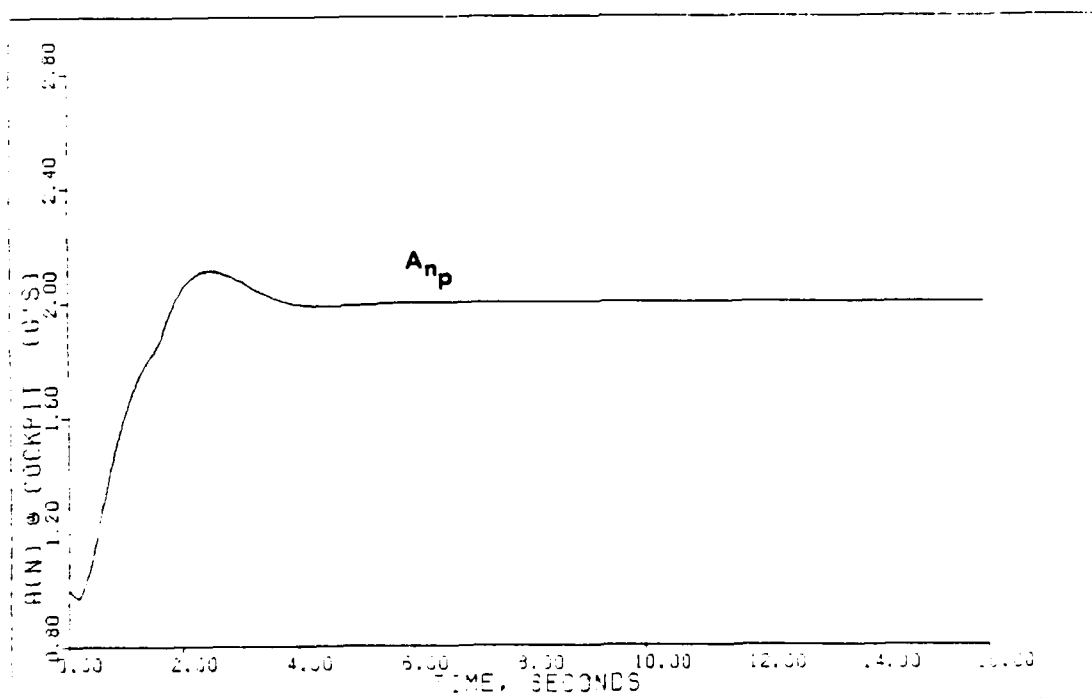
2G PULL-UP: PLANT+ACTUATORS+DELAY+SENSORS (2.0M/FL400)

Fig. 5.72



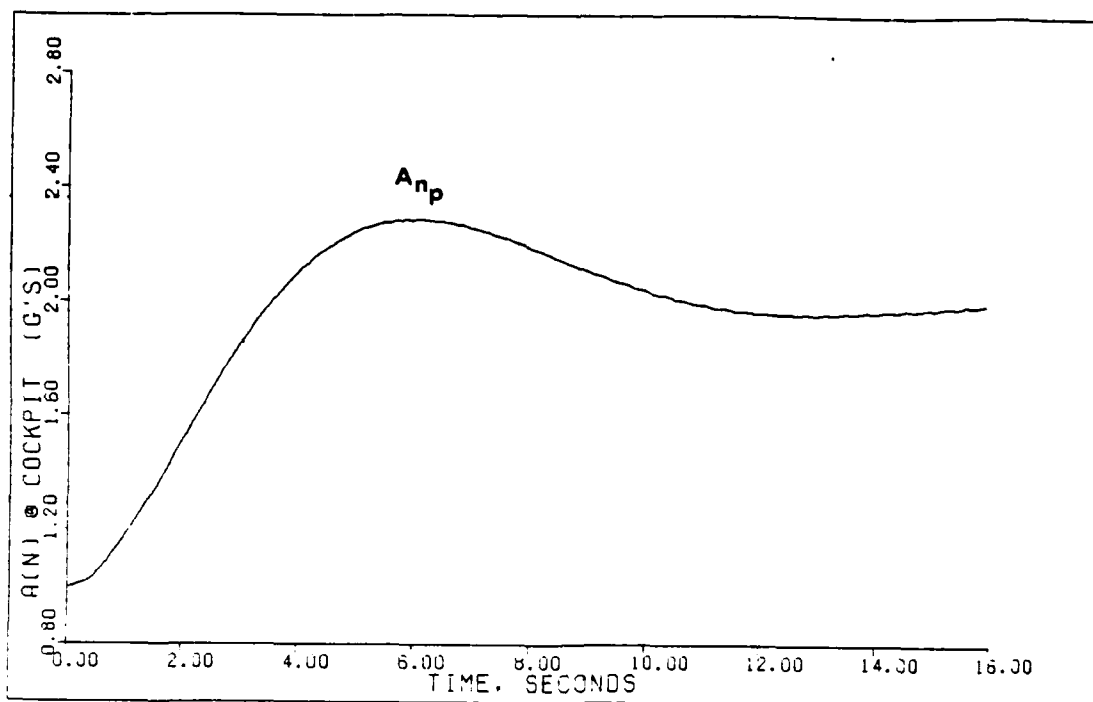
2G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.73



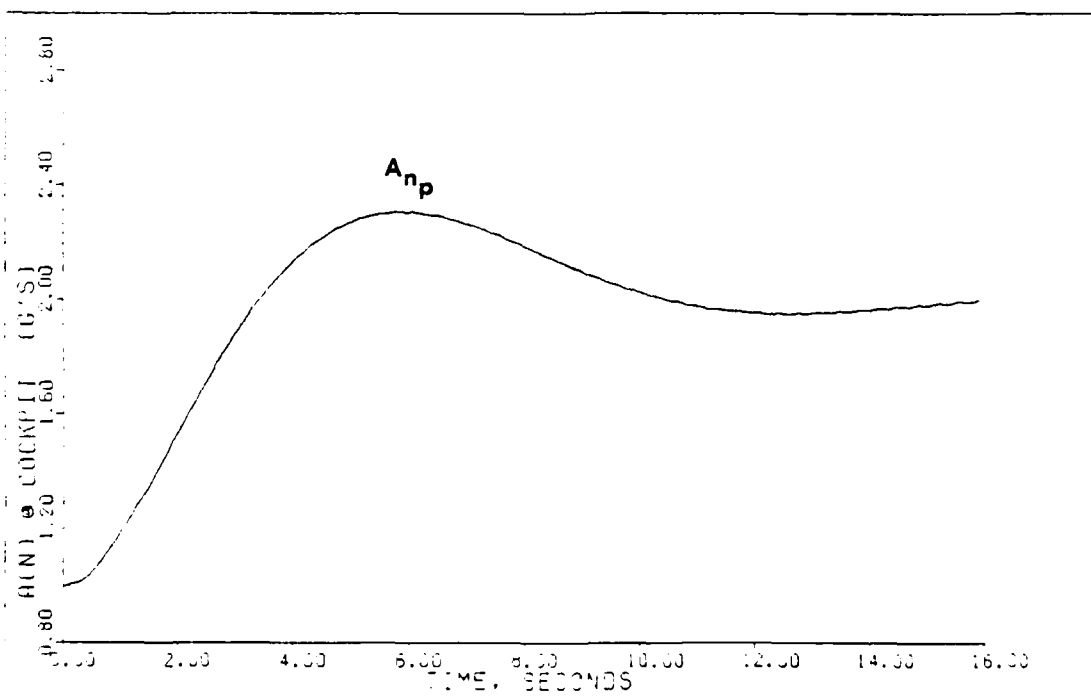
2G PULL-UP: PLANT-REACTORS (2.0M/FL400)

Fig. 5.74



2G PULL-UP: PLANT+ACTUATORS+DELAY (2.0M/FL400)

Fig. 5.75



2G PULL-UP: PLANT+ACTUATORS+DELAY+SENSORS (2.0M/FL400)

Fig. 5.76

Overall, the results from the 2 g command are excellent, especially when considering that A_{n_p} is never directly commanded. An attempt was made to eliminate the oscillation in A_{n_p} by smoothing the ramp input in theta. MULTI was modified to allow for this custom input feature by Acker (1) in a parallel thesis. This modified input did not improve the acceleration response. The remaining 2 g designs for 0.9 and 1.4 Mach are included in Appendix D. The results are similar with a more pronounced oscillation in A_{n_p} at the 0.9 Mach flight condition.

9.0 g Command Input. The 9 g maneuver demonstrates the capability of the control inputs in sustaining a maximum aircraft g loading. Obviously, the validity of the linear model is degraded since the assumptions of steady air flow and small perturbations can no longer be made. Since the purpose of this study is to examine the design and simulation of control laws formulated by the Porter method, this series of simulations is valuable and worth presenting for analysis.

Figure 5.77 shows the smooth and controlled input of the canard and stab-nozzle as the g loading is increased. The stab-nozzle is now the primary input for rotating the aircraft, with the canard rapidly swinging from a positive to negative deflection to balance the force and moment. Theta still tracks theta command with a delay that is

TABLE 5.9
DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Constant g Pull-Up (9.0 g's)

Flt Condition: 2.0 Mach at FL 400

Command Vector \underline{v} : $v_1 = \text{Theta: } 20, 2.653044, 20, 20$
 $v_2 = \text{Alpha: } 1.5, 0.1623156, 20, 20$

<u>Basic Plant</u>				
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
10.00	0.200	1.0	.2663E+01	-.6866E+00
		0.0025	-.6063E+00	-.1527E+00

Notes:

1. Each \underline{v} input is composed of four parts:
 - A. Time (secs) that the input reaches steady-state.
 - B. Steady-state value (radians).
 - C. Time (secs) input leaves steady-state.
 - D. Time (secs) input reaches zero.
2. Sigma = the elements (in order) of the diagonal matrix.
3. The integral controller matrix $\underline{K}_1 = (\text{alpha})\underline{K}_0$.
4. Irregular Design: $\underline{M} = (0.3, 0)^T$.

TABLE 5.10
DESIGN OUTPUT FIGURES OF MERIT

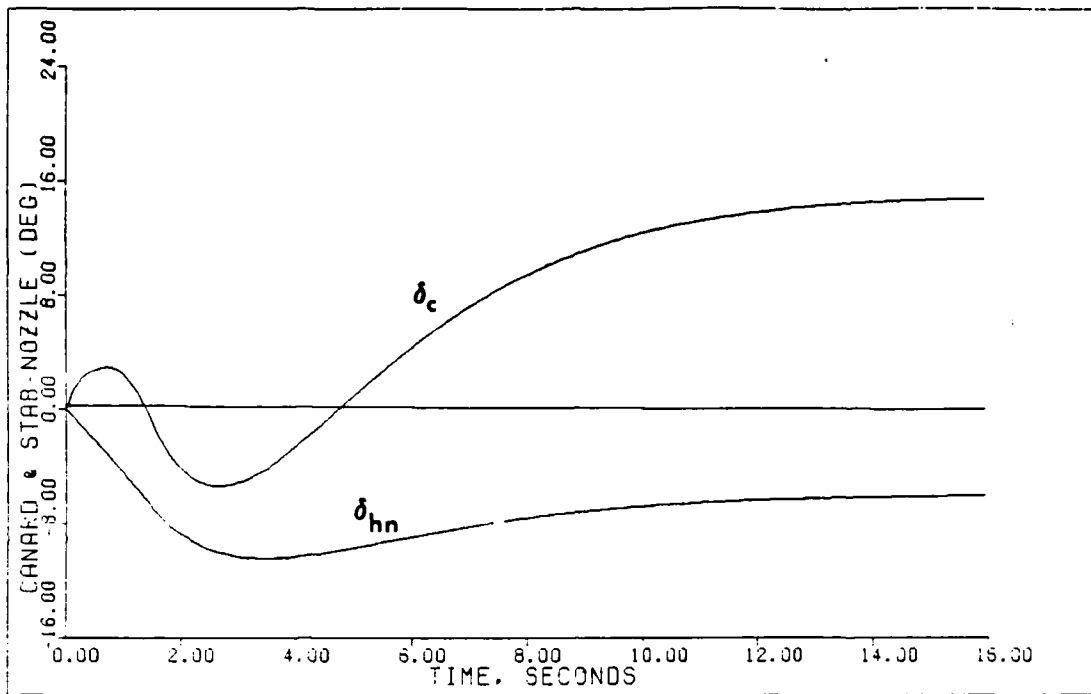
Maneuver: Constant g Pull-Up (9.0 g)

Flt Condition: 2.0 Mach at FL 400

<u>Basic Plant</u>			
<u>Output</u>	<u>Peak Value</u>	<u>Peak Time</u>	<u>Settling Time</u>
Acceleration	+9.0	N/A	4.8
Angle of Attack	+10.03	5.25	10.5
Pitch Rate	+7.95	1.5	2.5

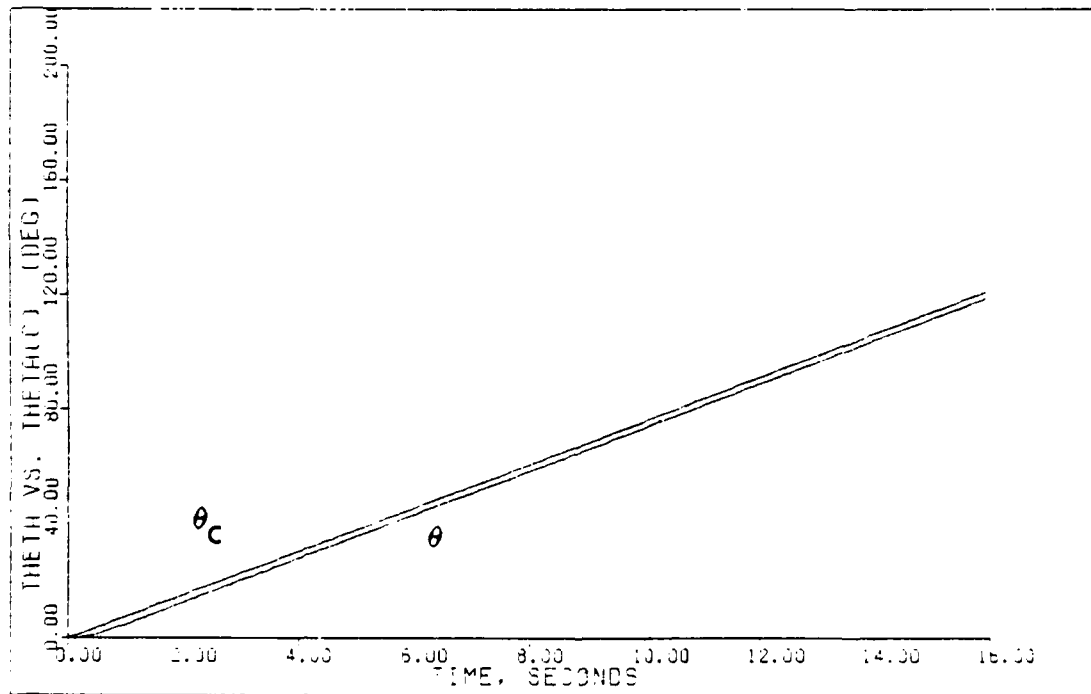
Notes:

1. See Table 5.9 for the command vector.
2. The final value of angle of attack equals the commanded step input (integral control).
3. All values for acceleration and pitch rate are estimated from the response plots.
4. Units for alpha are in degrees, time in seconds, pitch rate in degrees/second, and acceleration in g's.



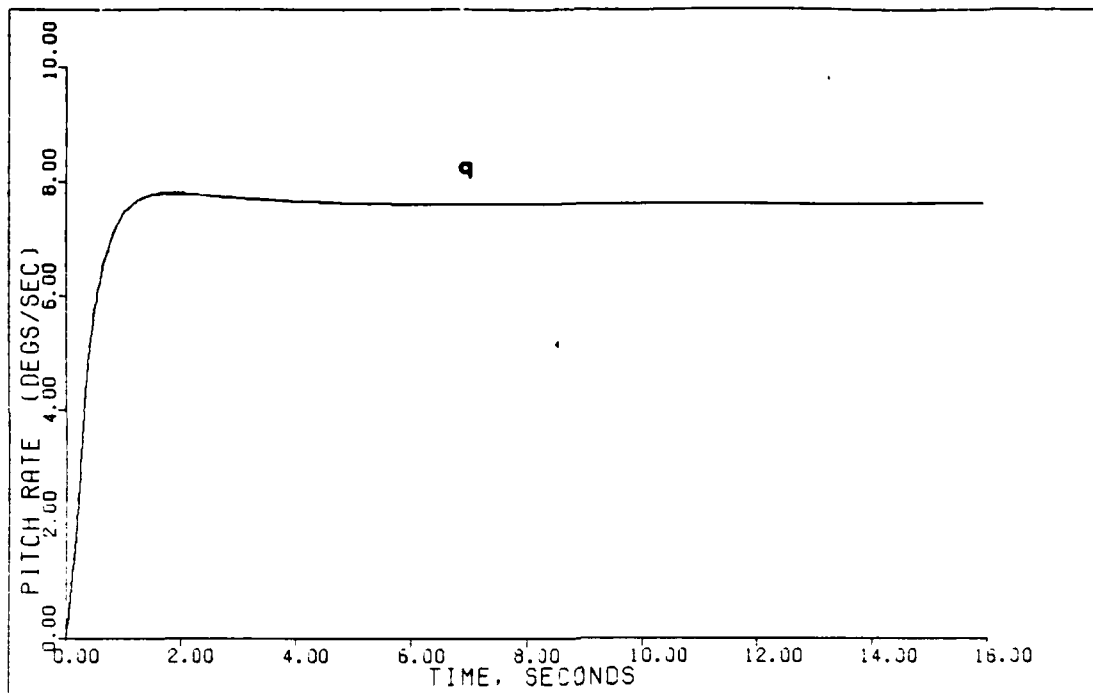
9G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.77



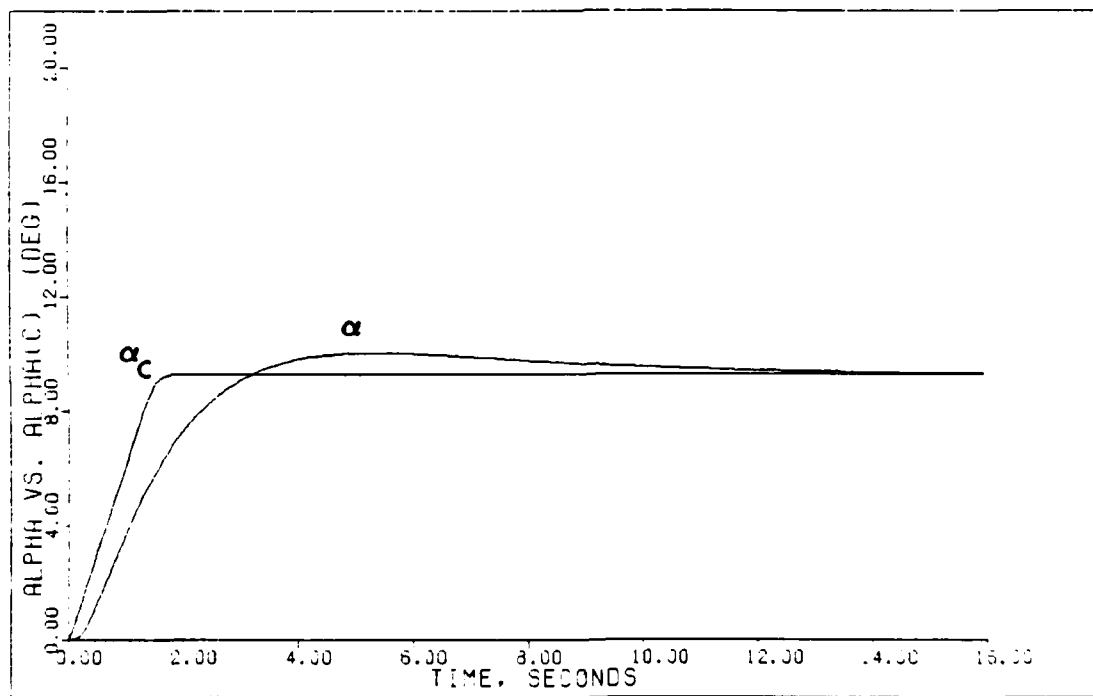
9G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.78



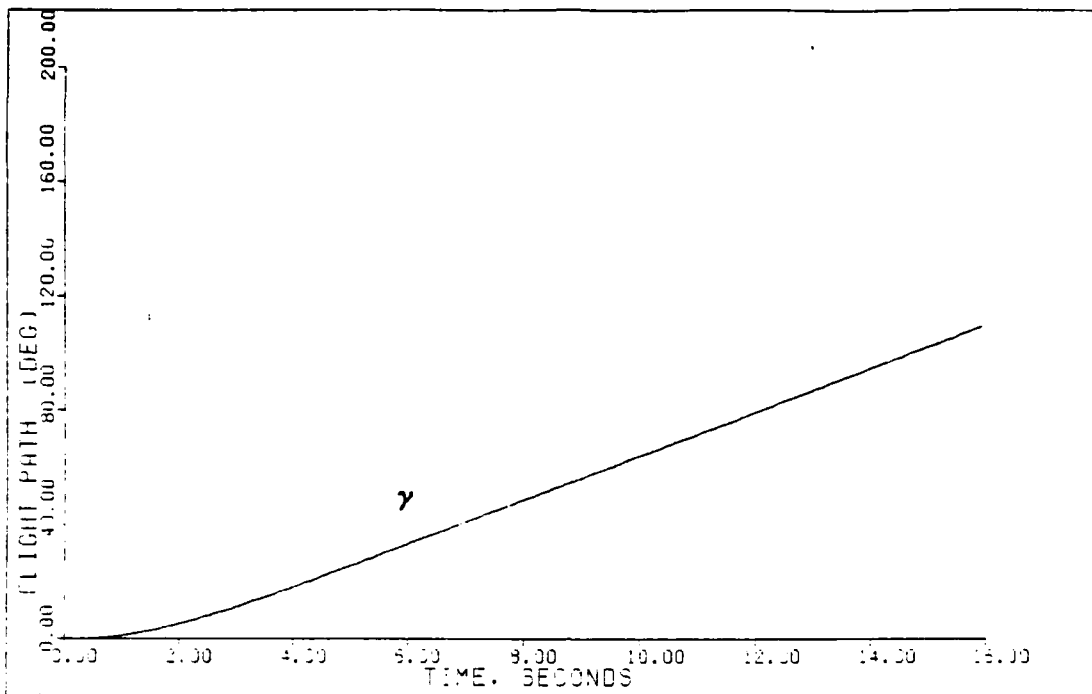
9G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.79



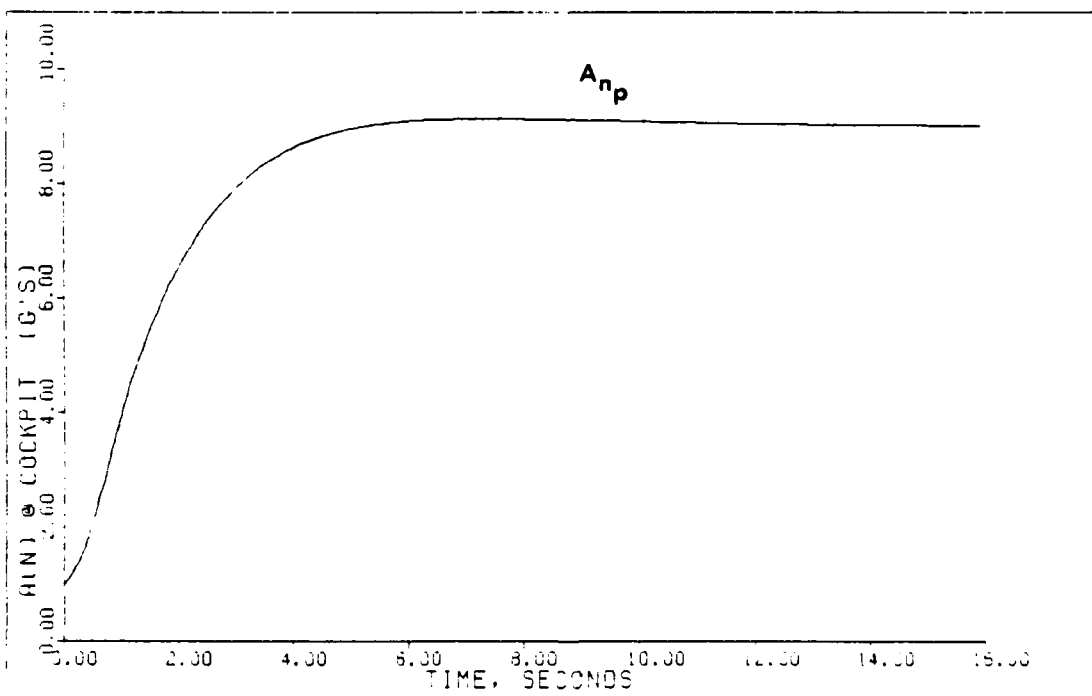
9G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.80



9G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.81



9G PULL-UP: BASIC PLANT (2.0M/FL400)

Fig. 5.82

larger than in the 2 g simulation. This error is directly proportional to the magnitude of the input and inversely proportional to the forward loop gain. The 9 g controller gain is smaller ($\bar{\alpha} \in \Sigma$) than in the 2 g design which, by itself, creates a larger error (Tables 5.8 and 5.10). As before, since q is the commanded input, the error in theta is not apparent to the pilot.

Pitch rate is a smooth, "first-order" type response with a rise time of 0.95 secs and a settling time of 2.5 secs (Figure 5.79). This rapid settling time provides for a very controlled g onset as sensed by the pilot (Figure 5.82). The remaining 9 g simulations for the two remaining flight conditions are included in Appendix D. The results of these simulations are very similar to those presented in this chapter.

5.6 Parameter Variation Results

The results of parameter variation are divided into two separate subsections to provide greater clarity. As mentioned previously, the single controller analysis is not, strictly speaking, a demonstration of parameter variation. However, since it does demonstrate the controller's flexibility in responding to different input commands, it is included in this section.

Single Controller Analysis. The first task in the parameter variation problem is to come up with a single

controller at each flight condition that will perform all of the maneuvers with satisfactory results. Because of time constraints, this is accomplished at two of the four flight conditions: 1.4 Mach/FL 200 and 2.0 Mach/FL 400. The constant g maneuver uses the reduced order three-state model and therefore is not included in this analysis.

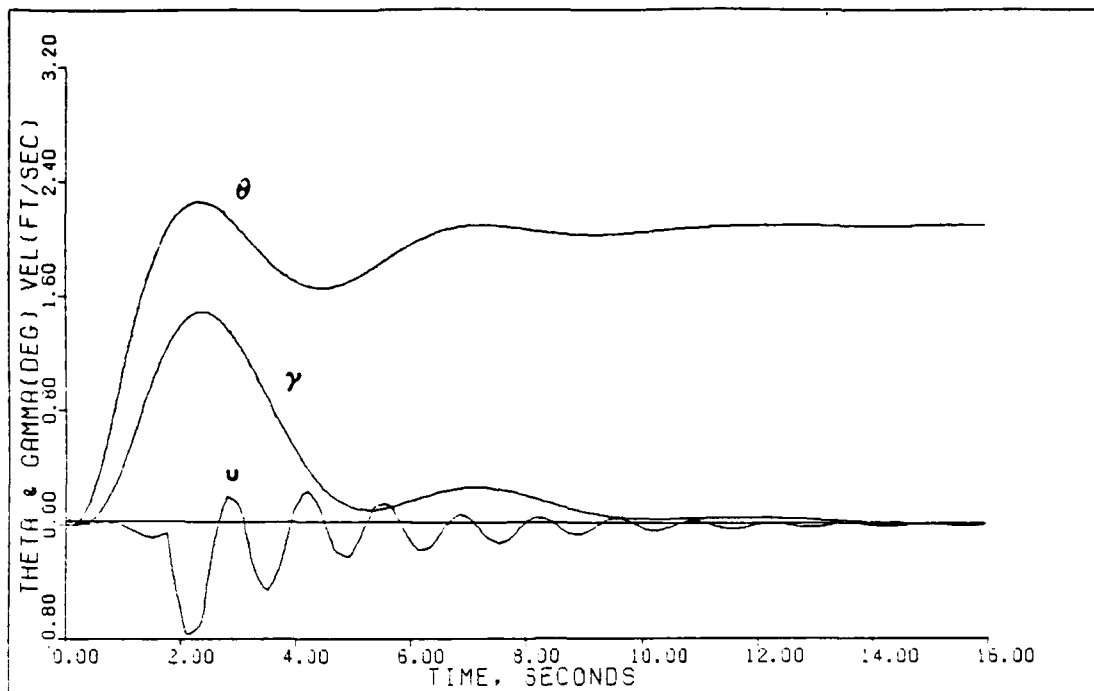
Figures 5.83 through 5.85 show the maneuver responses using the direct climb controller designed at 1.4 Mach/FL 200. These simulations include actuators, computational time delay, and sensors. The pitch pointing maneuver shows larger transients in both flight path and velocity when compared with the results from its own controller (Figure 5.48). In addition, the velocity settling time has increased dramatically. By comparing the design parameters for the two controllers (Tables 5.1 and 5.5), the lower σ_3 value in the direct climb controller accounts for the degraded performance in flight path. This condition cannot be changed since any increase in σ_3 causes the canard to overshoot its maximum deflection limit. A second-order oscillation has also developed in the pitch angle since the slightly higher integral gain has a destabilizing effect on the output response. The overall performance of the maneuver remains stable, however, with steady-state achieved in approximately 10 secs.

The vertical translation maneuver gives better results using the direct climb controller (Figure 5.84).

The oscillatory response in gamma is predicted because of the low σ_3 value; however, this result is similar to the vertical translation maneuver using its own controller (Appendix D). As with pitch pointing, the vertical translation maneuver is stable throughout the simulation and reaches steady-state within 10 secs.

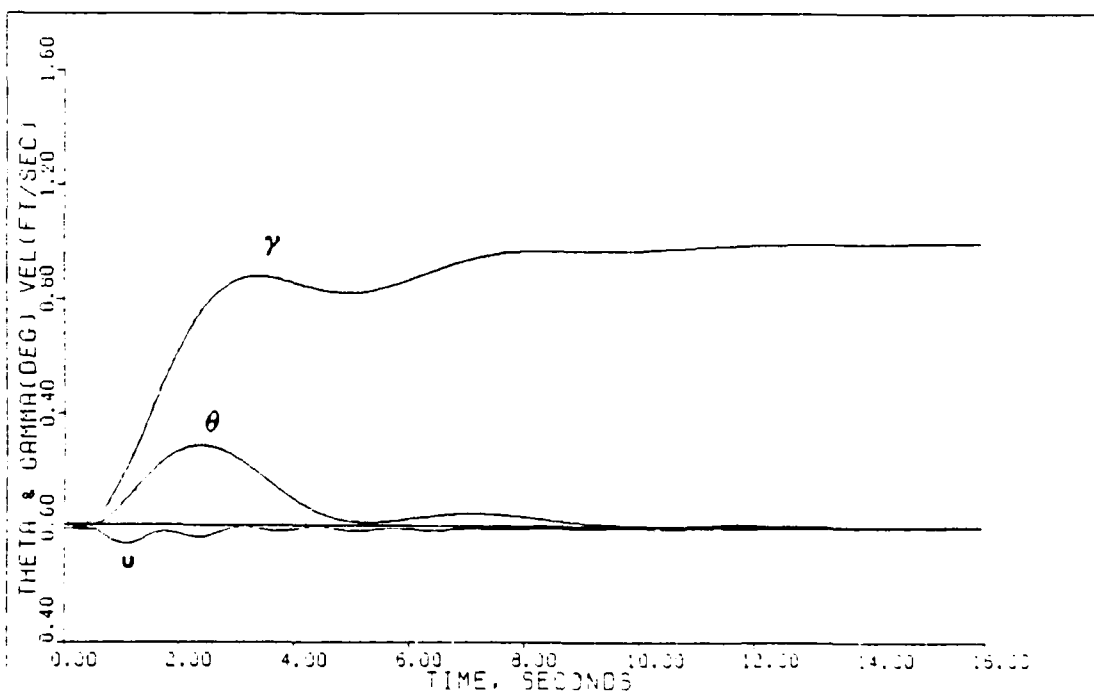
At the 2.0 Mach/FL 400 flight condition, the direct climb controller is again chosen because of its design parameters. The σ_3 value must be below 0.02 to prevent excessive canard deflections in the direct climb and vertical translation maneuvers. This restriction eliminates the pitch pointing controller. The gain value (epsilon) in the vertical translation design would cause instability during a direct climb. Consequently, the direct climb controller offers the best compromise in design values. Figures 5.86 through 5.88 present the single controller results at this design point. Once again, the large transient in gamma during the pitch pointing results from the low σ_3 value corresponding to flight path. The similarity between direct climb and vertical translation designs gives excellent results in the vertical translation simulation (Figure 5.88).

Overall, the single controller analysis provides stable, well behaved results for the flight conditions tested. In many cases, however, performance is reduced for the benefit of simplicity. A single controller that



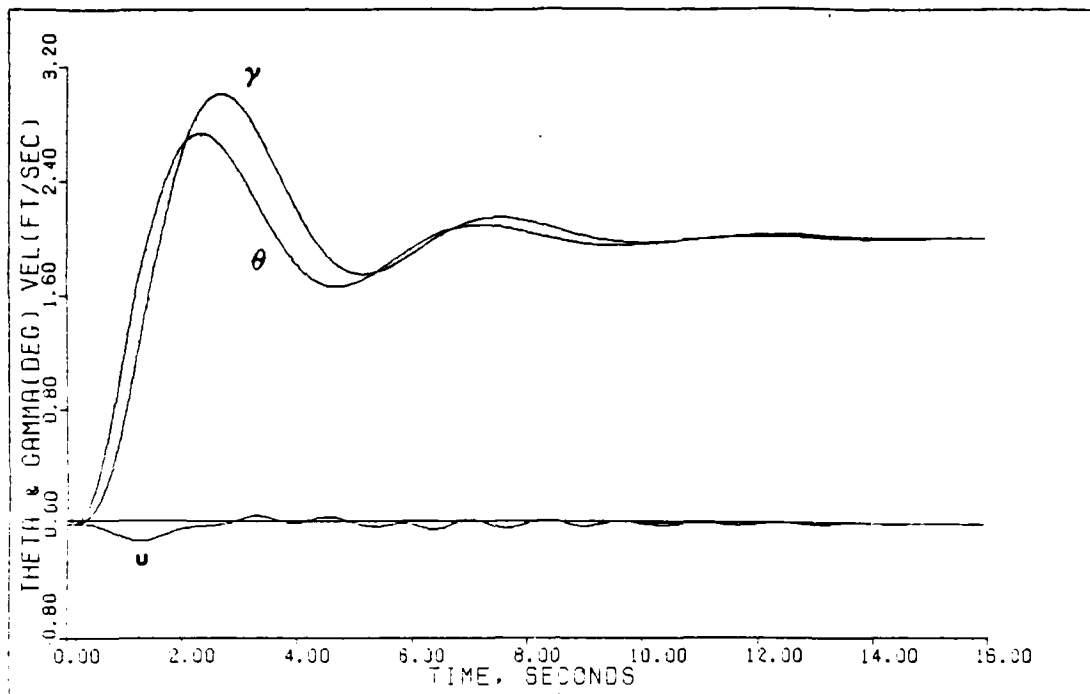
PITCH POINTING: SINGLE CONTROLLER ANALYSIS (1.4M/FL200)

Fig. 5.83



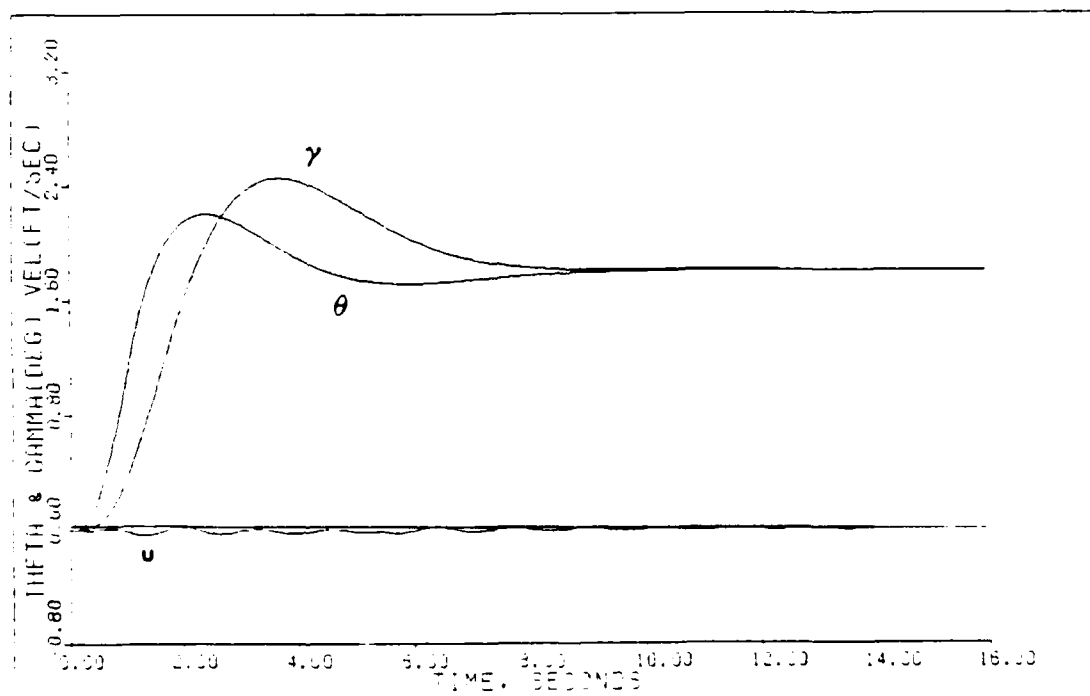
VERTICAL TRANS: SINGLE CONTROLLER ANALYSIS (1.4M/FL200)

Fig. 5.84



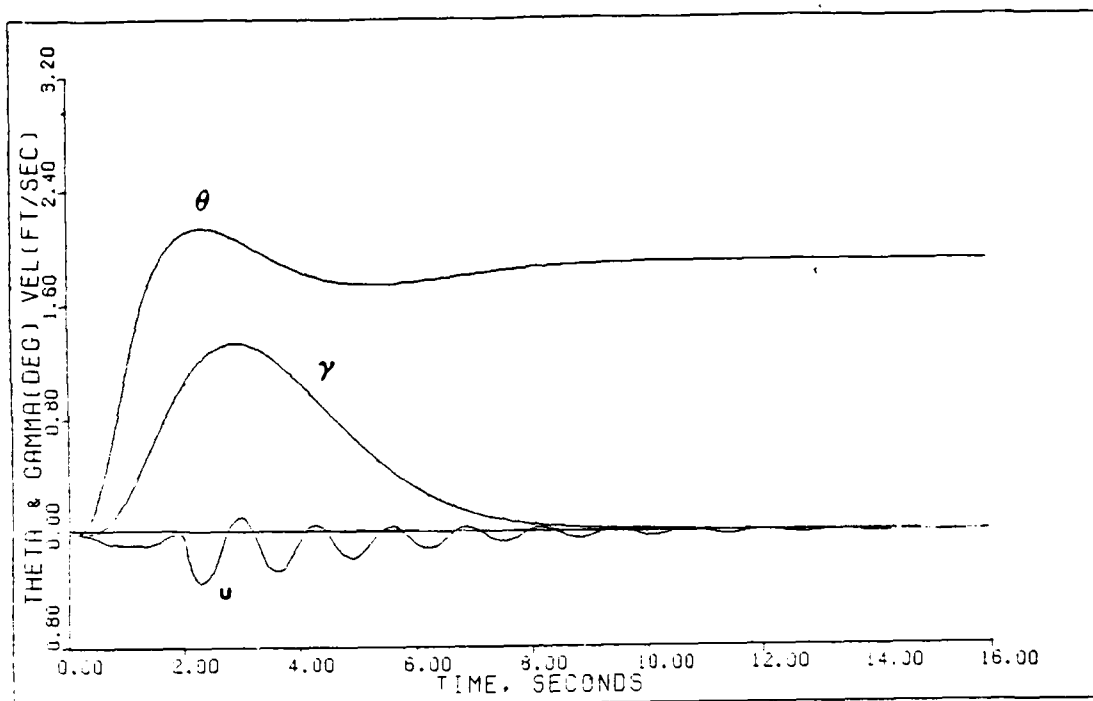
DIRECT CLIMB: SINGLE CONTROLLER ANALYSIS (1.4M/FL200)

Fig. 5.85



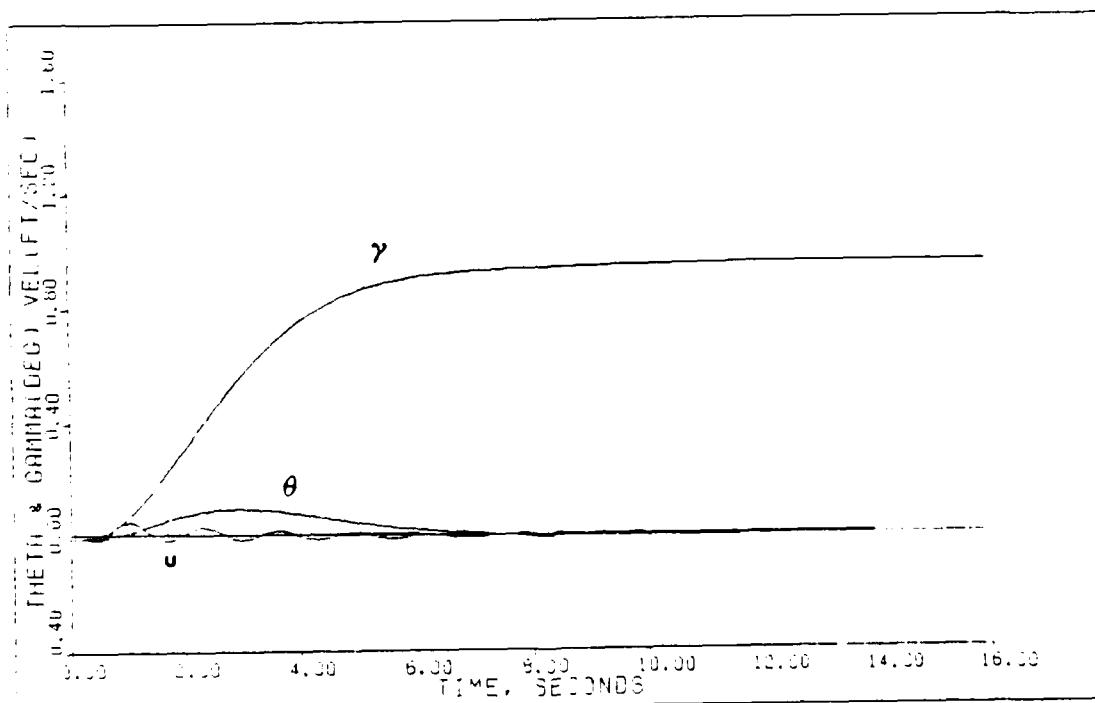
DIRECT CLIMB: SINGLE CONTROLLER ANALYSIS (1.0M/FL400)

Fig. 5.86



PITCH POINTING: SINGLE CONTROLLER ANALYSIS (2.0M/FL400)

Fig. 5.87



VERTICAL TRANS: SINGLE CONTROLLER ANALYSIS (2.0M/FL400)

Fig. 5.88

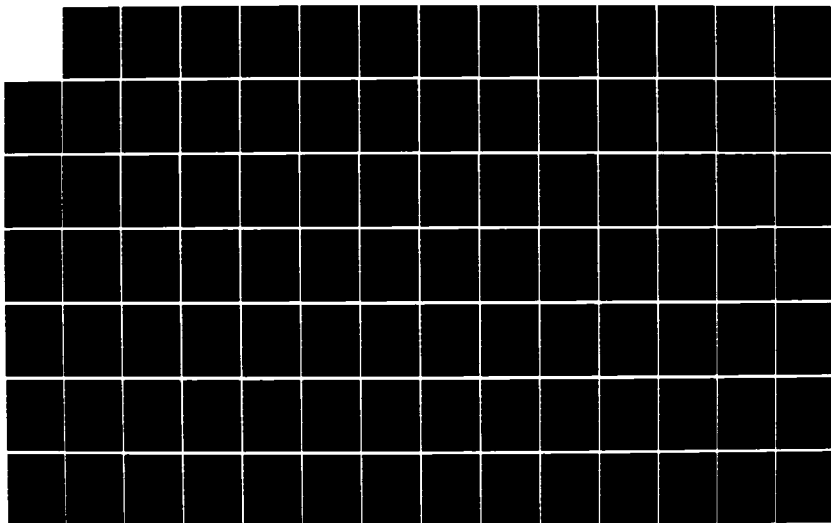
AD-A164 817

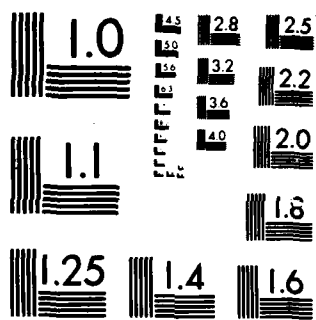
MULTIVARIABLE CONTROL LAW DESIGN FOR ENHANCED AIR
COMBAT MANEUVERING F-15. (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. K A SNEEMAN
DEC 85 AFIT/GE/EE/85D-38 F/G 1/2

3/5

UNCLASSIFIED

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

is gain scheduled between design points is very desirable, from a pilot's standpoint, since it eliminates having to reconfigure gains prior to commanding the maneuver. In a combat application, performance would have to be sacrificed in favor of a single controller.

Controller Sensitivity. As explained in Chapter IV, a measure of control robustness is the insensitivity that it exhibits to variations in the model coefficients. The most important of these coefficients is the control derivatives since minor changes can result in system instability.

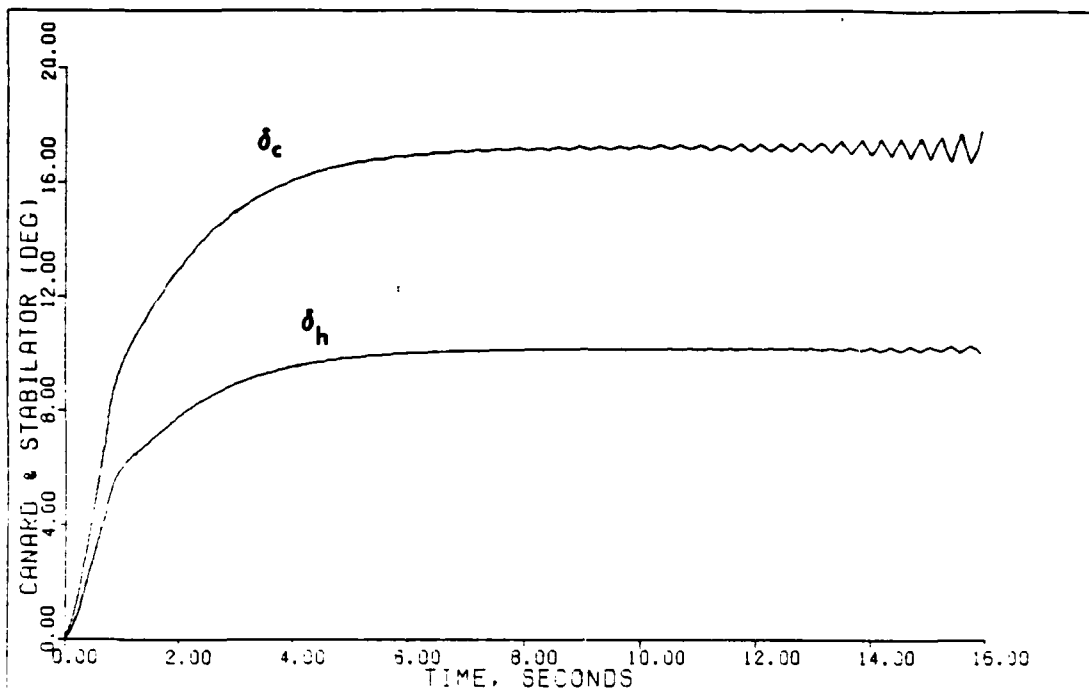
The results of the sensitivity analysis demonstrate excellent controller capacity for handling parameter variations. Figures 5.89 through 5.92 show the effects of a +9 percent change in M_{δ_c} on the system response. This maximum value is found by slowly increasing the selected derivative until instability is present in any of the system responses. Instability occurs first in the canard. Its effect, however, is quickly "felt" in the other control inputs. Because of the low-pass frequency characteristics of the system, the input instability is not present in the system output.

A reduction in M_{δ_c} , even by the smallest amount, causes control input instability beyond the 12 sec point in the simulation. By adjustment of the controller gain, this instability could be eliminated, providing for a more

symmetrical variation allowance in M_{δ_c} . Once again, the control input divergence is not passed to the system outputs.

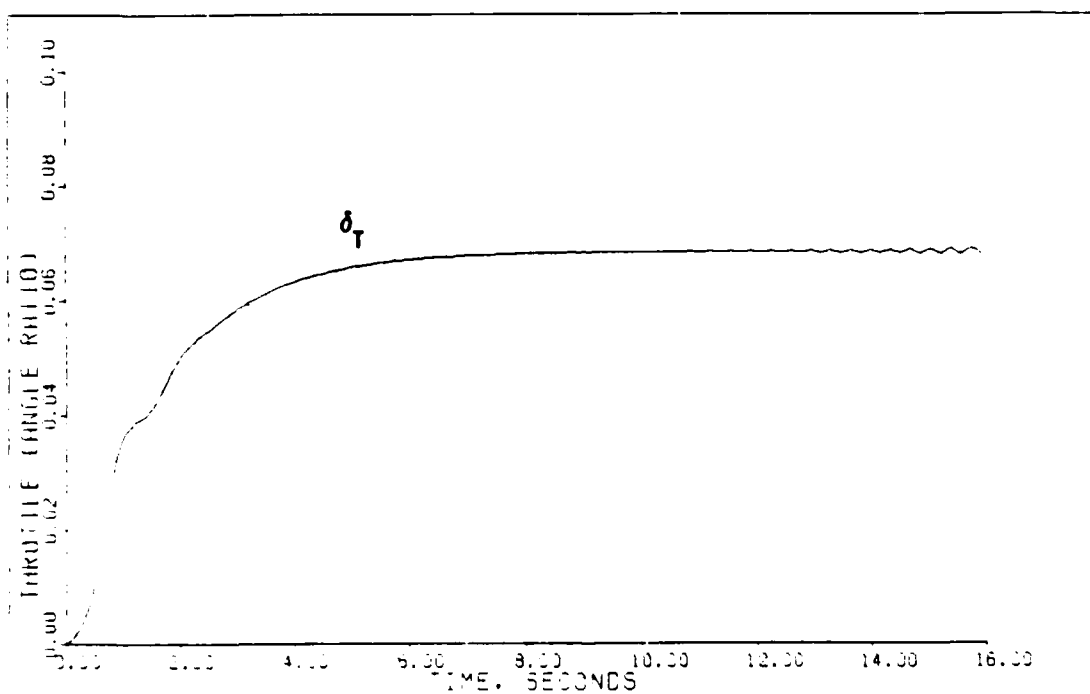
The frequency of this divergence is somewhat distorted by the plotting function. Only 100 of the total 640 data points (16 secs at 40 Hz sampling frequency) are used for plotting. The actual frequency of the output data is 20 Hz, or half of the sampling frequency. It is well known that the highest frequency content of any sampled data system output can never be higher than half the sampling frequency (12). The 20 Hz divergence is not apparent, however, because of the plotting function.

A similar adjustment is made to M_{δ_H} and the results are presented in Figures 5.97 through 5.104. As the derivative is increased by +7 percent, the stabilator is now more effective w.r.t. the pitching moment and therefore deflects to a lower steady-state value. Control input instability results, however, at 12 sec into the simulation. The canard goes dynamically unstable again at a faster rate than the other inputs. This oscillation is isolated from the outputs but the new value for M_{δ_H} does affect their transient characteristics (Figures 5.99 and 5.103). When the derivative is reduced by -25 percent, control instability returns and the stabilator now deflects to a larger value on the steady-state (reduced effectiveness). Figures 5.101 to 5.104 demonstrate these results.



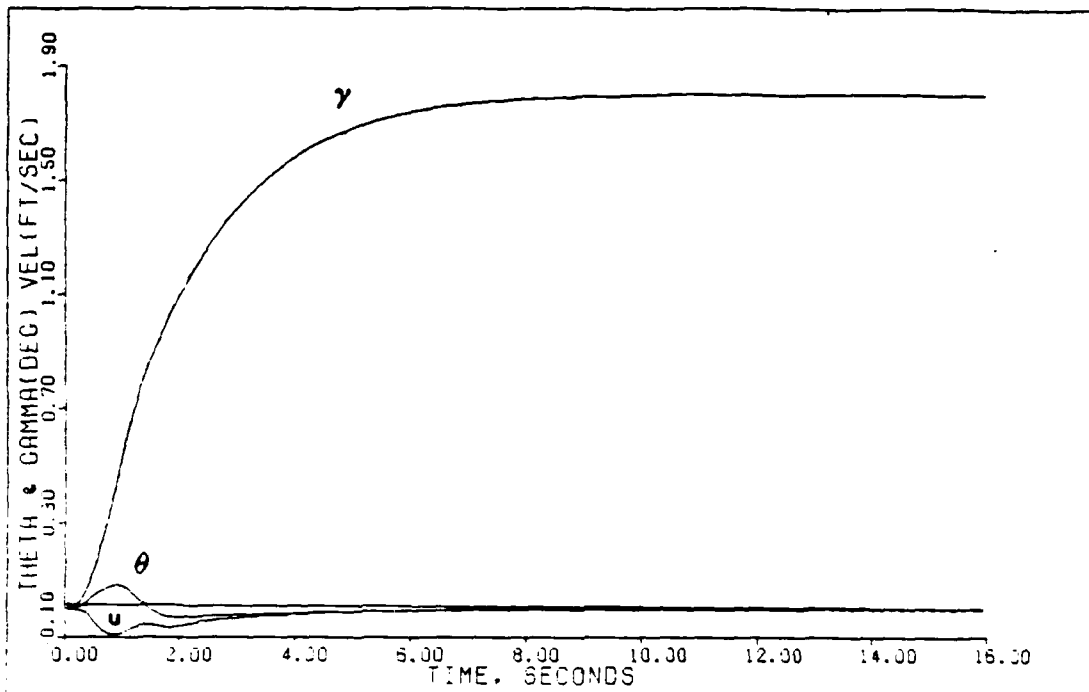
VERTICAL TRANS: +9% M-DELTA(0). (0.9M/FL200)

Fig. 5.89



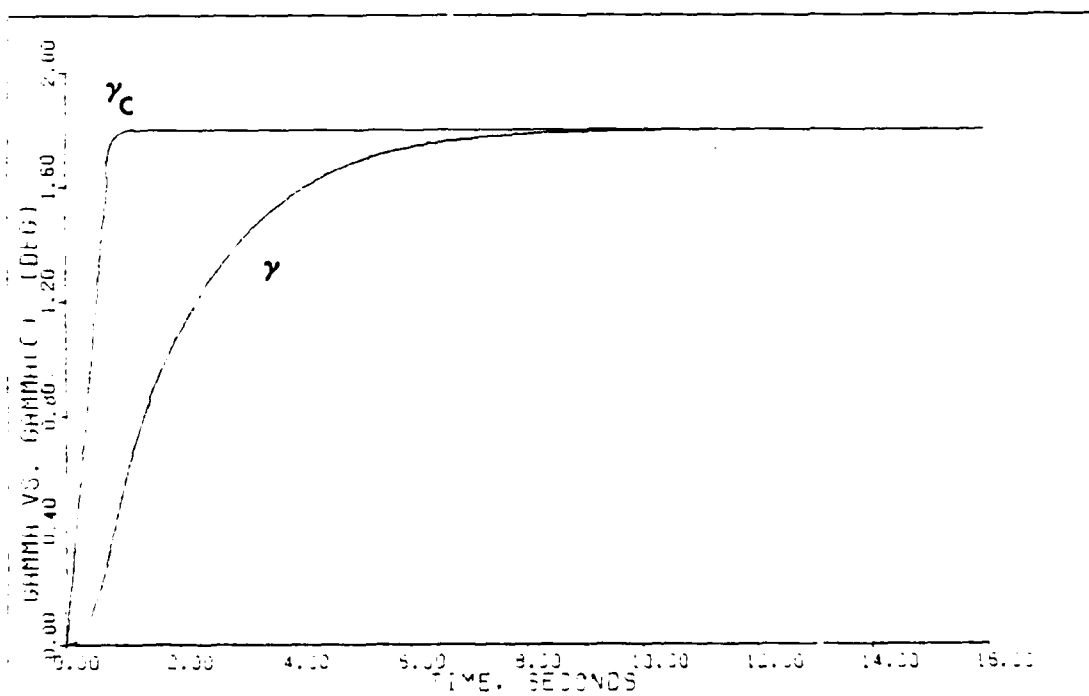
VERTICAL TRANS: +9% M-DELTA(0). (0.9M/FL200)

Fig. 5.90



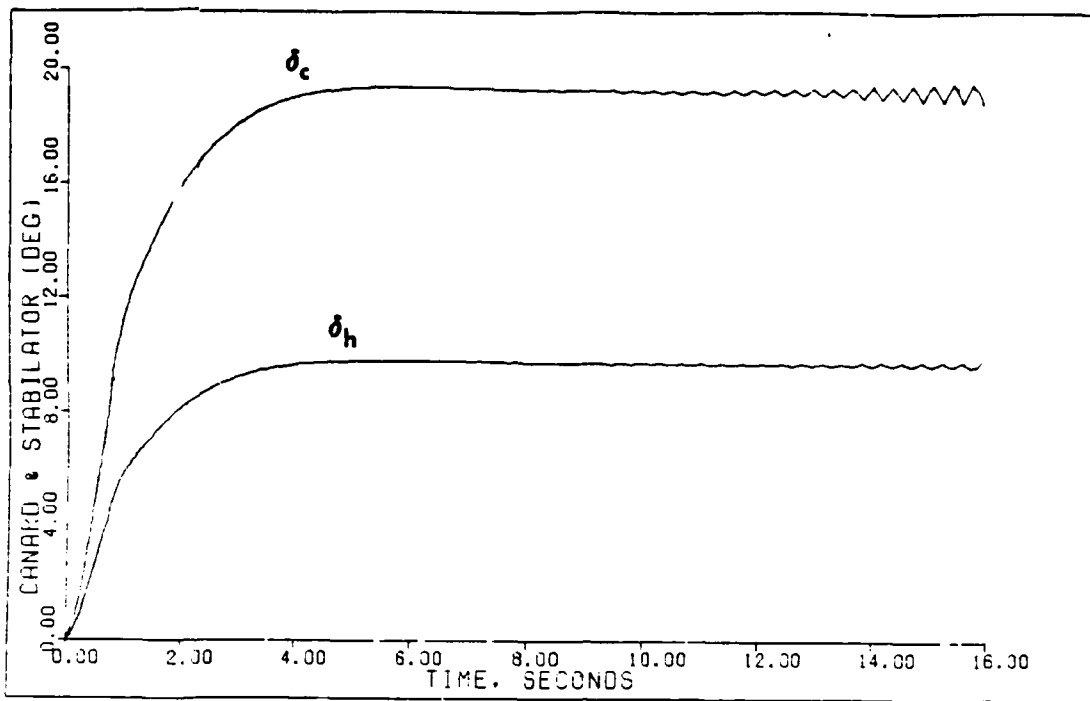
VERTICAL TRANS: +9% M-DELTA(C), (0.9M/FL200)

Fig. 5.91



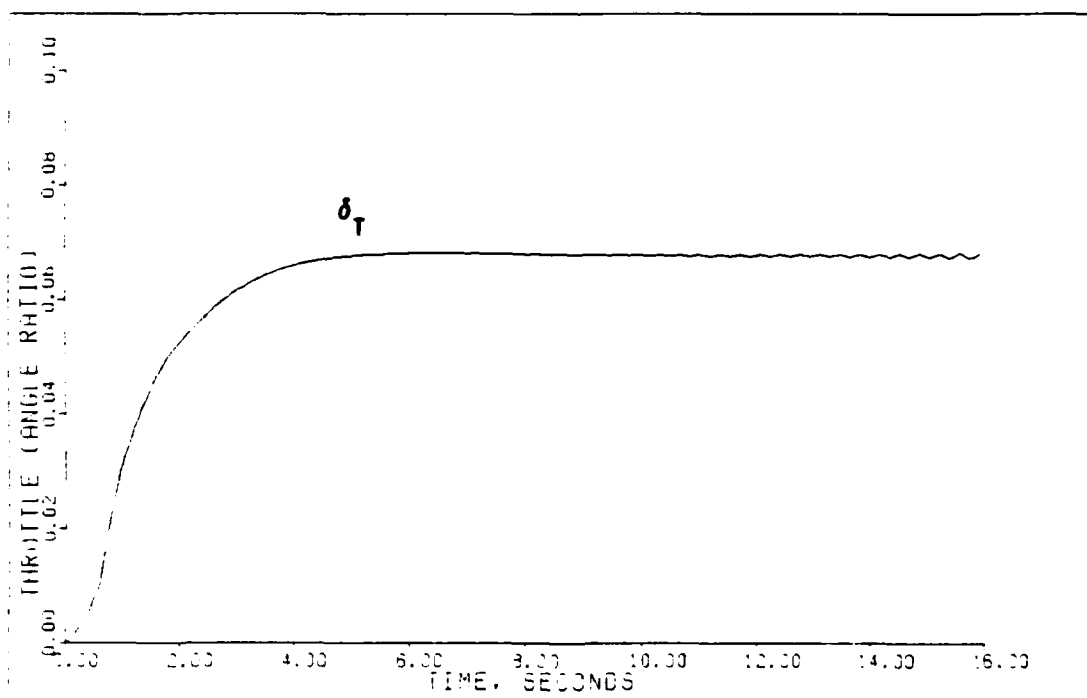
VERTICAL TRANS: +9% M-DELTA(C), (0.9M/FL200)

Fig. 5.92



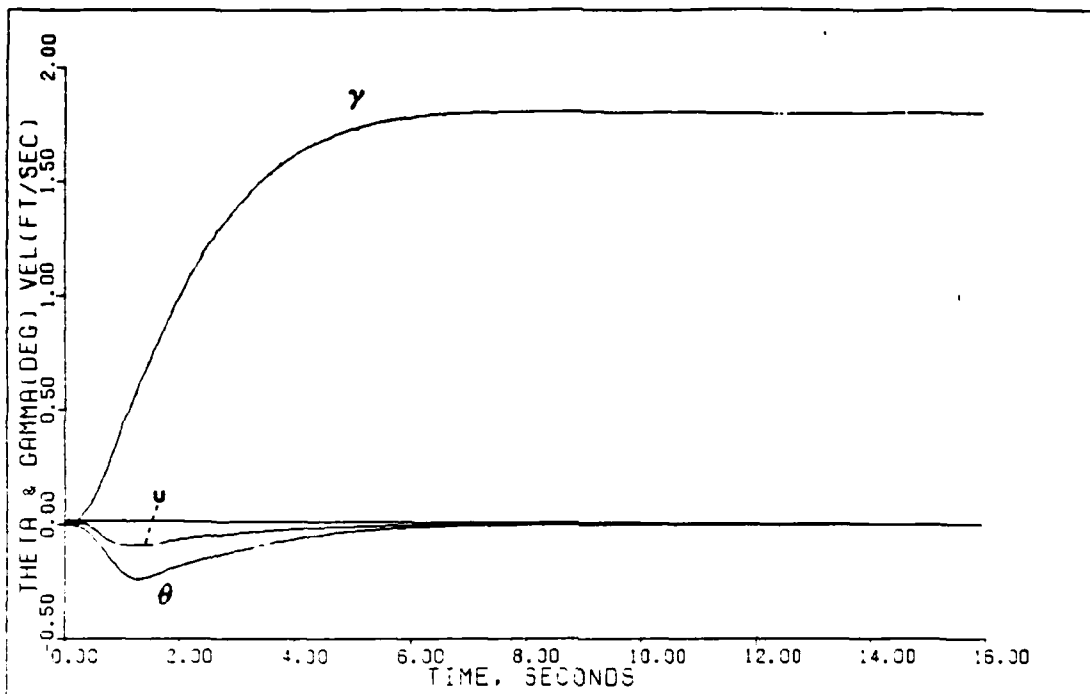
VERTICAL TRANS: -0.5% M-DELTA (C), (0.3M/FL200)

Fig. 5.93



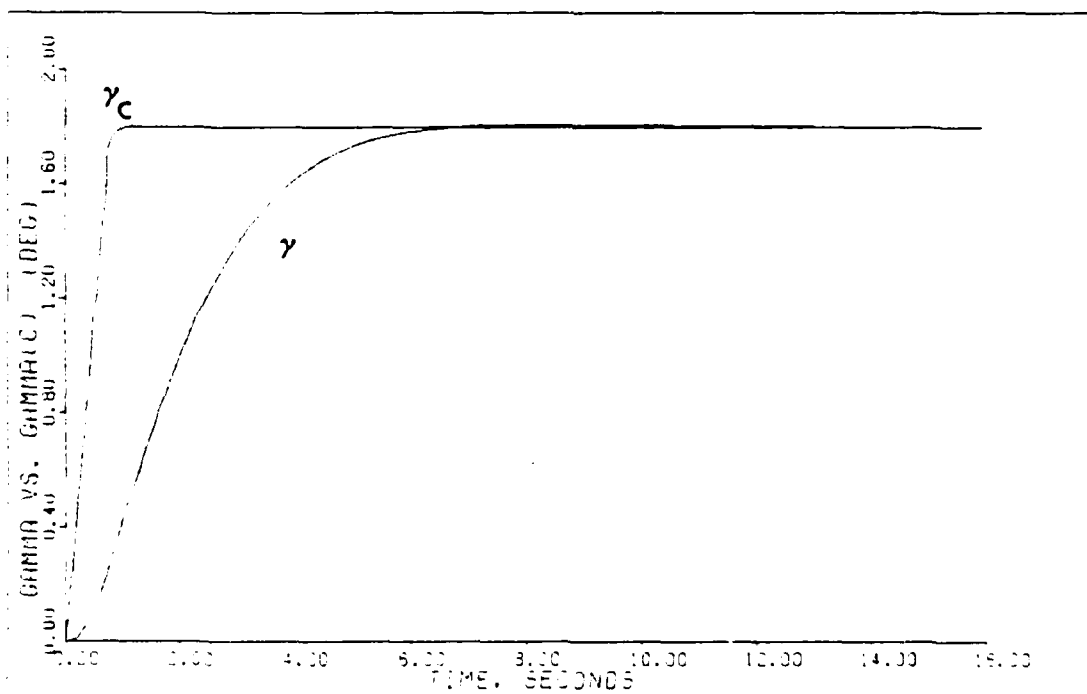
VERTICAL TRANS: -0.5% M-DELTA (C), (0.3M/FL200)

Fig. 5.94



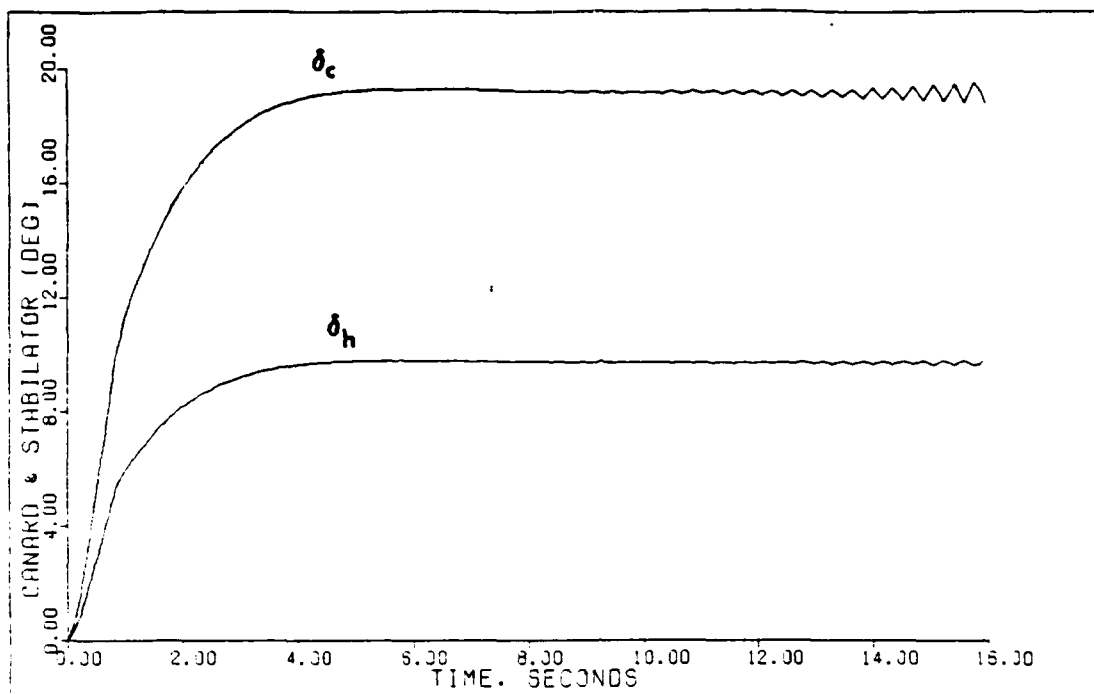
VERTICAL TRANS: -0.5% M-DELTA (C), (0.9M/FL200)

Fig. 5.95



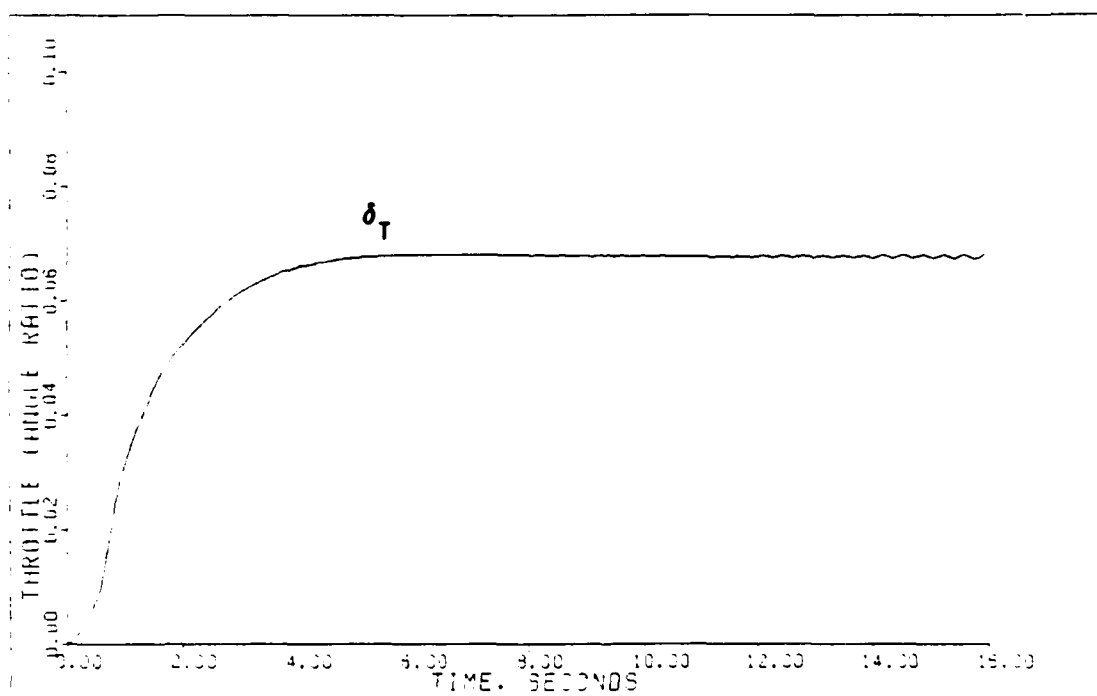
VERTICAL TRANS: -0.5% M-DELTA (C), (0.9M/FL200)

Fig. 5.96



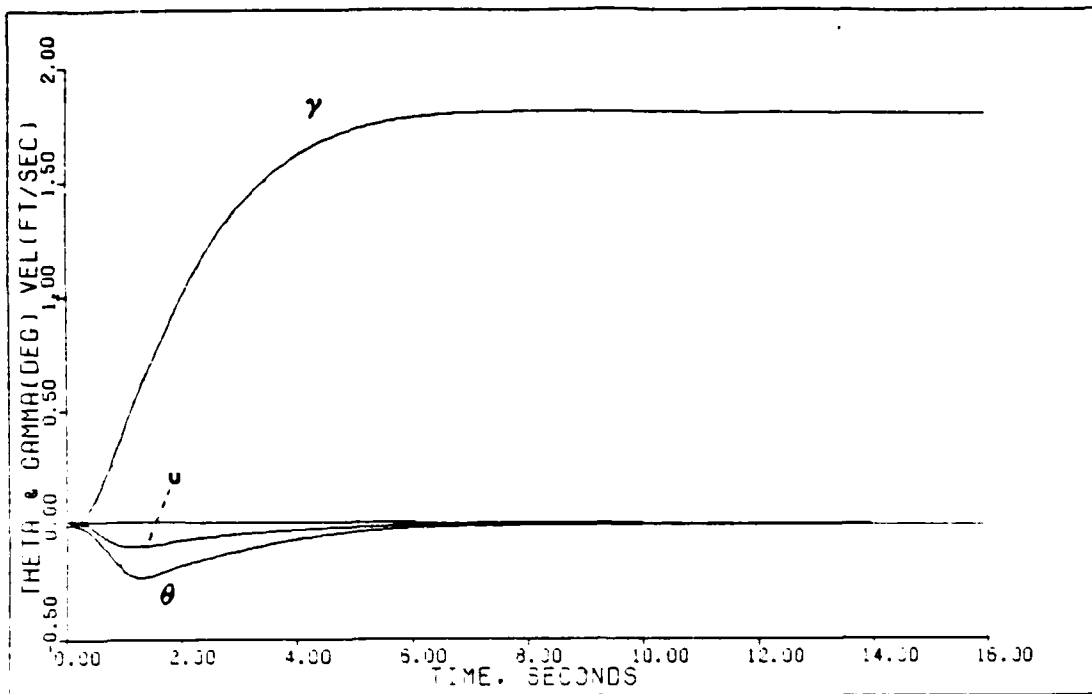
VERTICAL TRANS: +7% M-DELTA(H). (0.9M/FL200)

Fig. 5.97



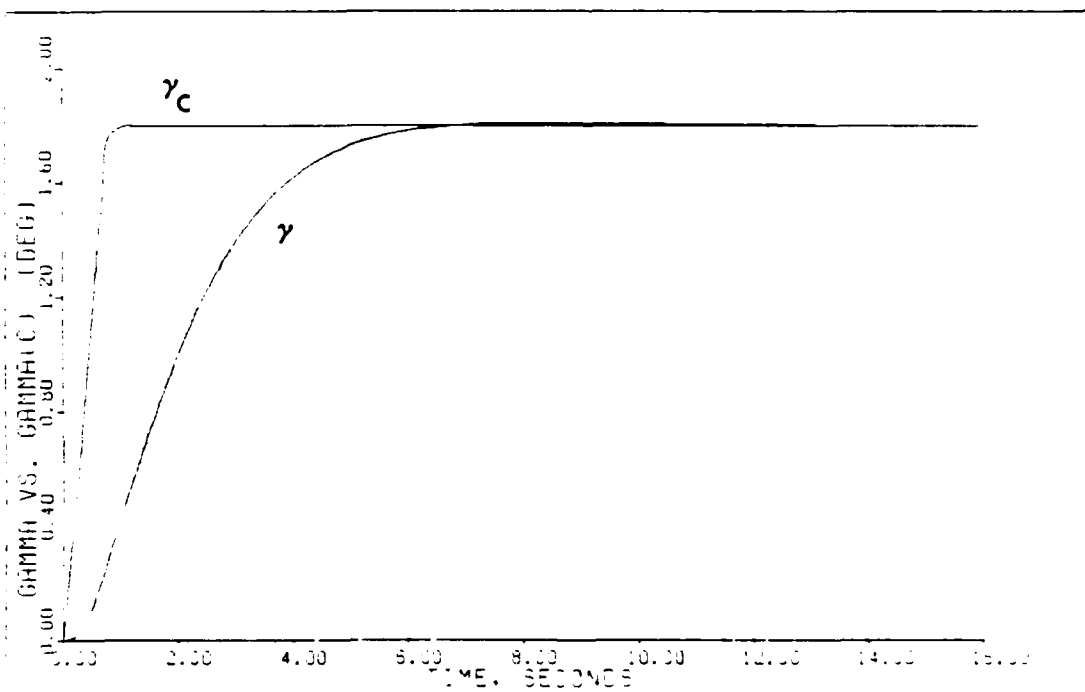
VERTICAL TRANS: +7% M-DELTA(H). (0.9M/FL200)

Fig. 5.98



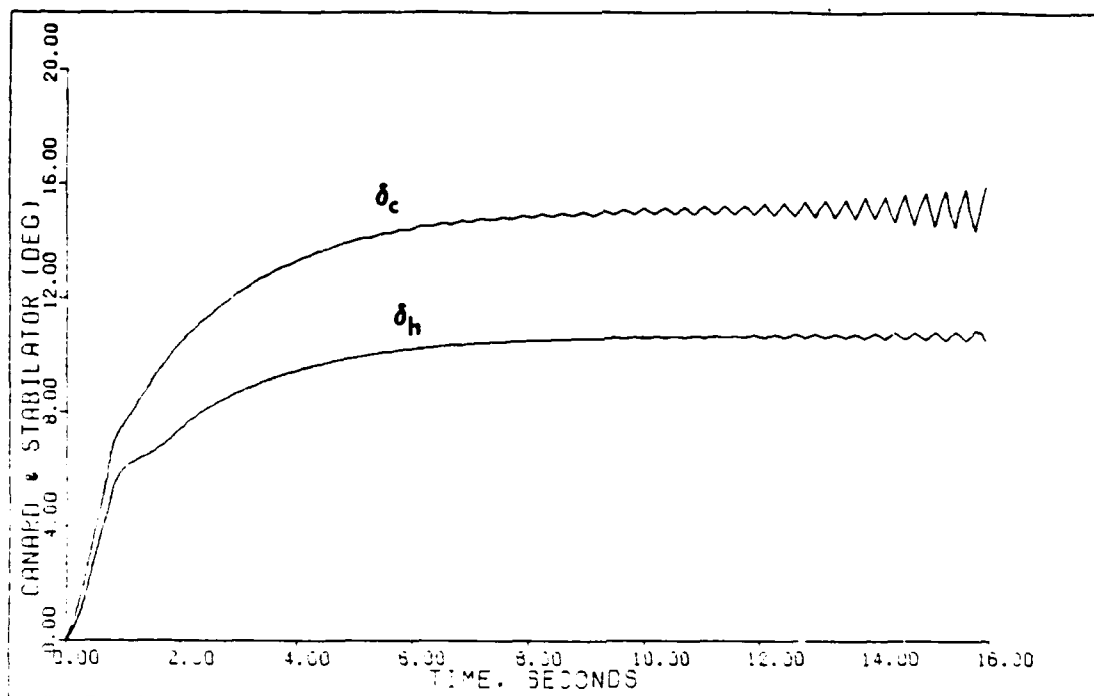
VERTICAL TRANS: +7% M-DELTA(H), (0.9M/FL200)

Fig. 5.99



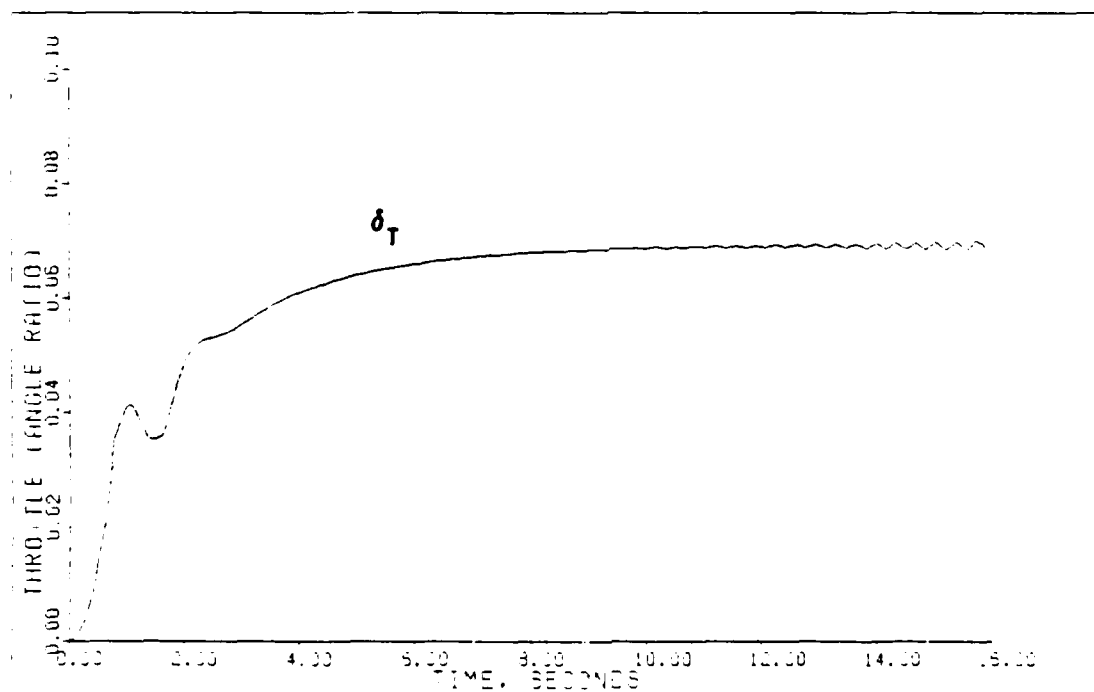
VERTICAL TRANS: +7% M-DELTA(H), (0.9M/FL200)

Fig. 5.100



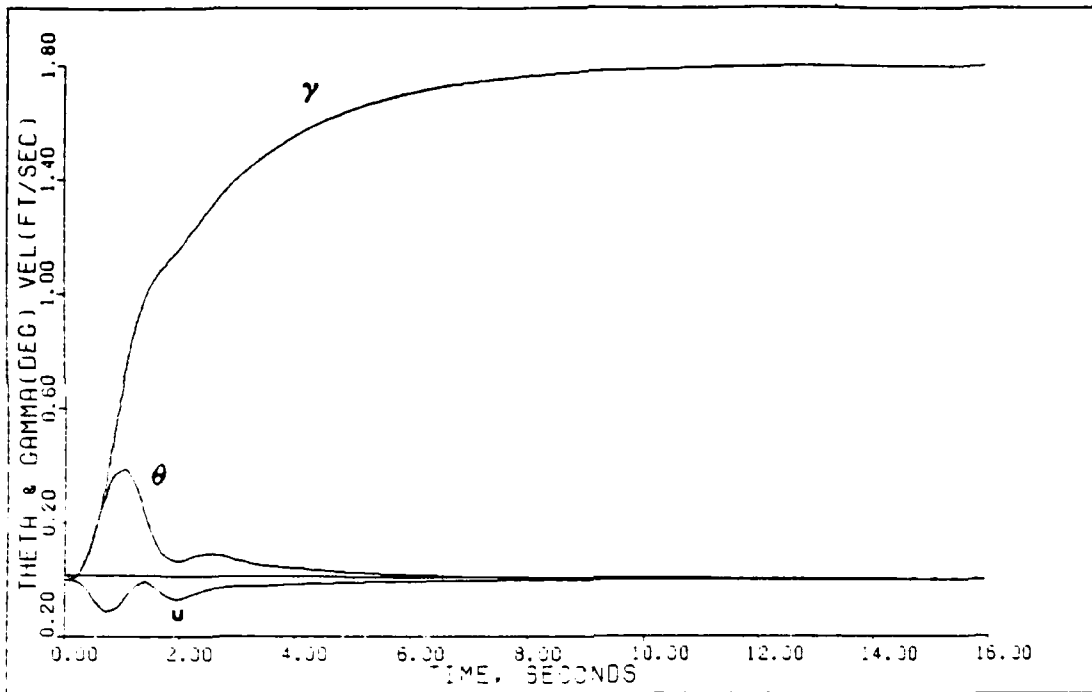
VERTICAL TRANS: -25% M-DELTA(H). (0.9M/FL200)

Fig. 5.101



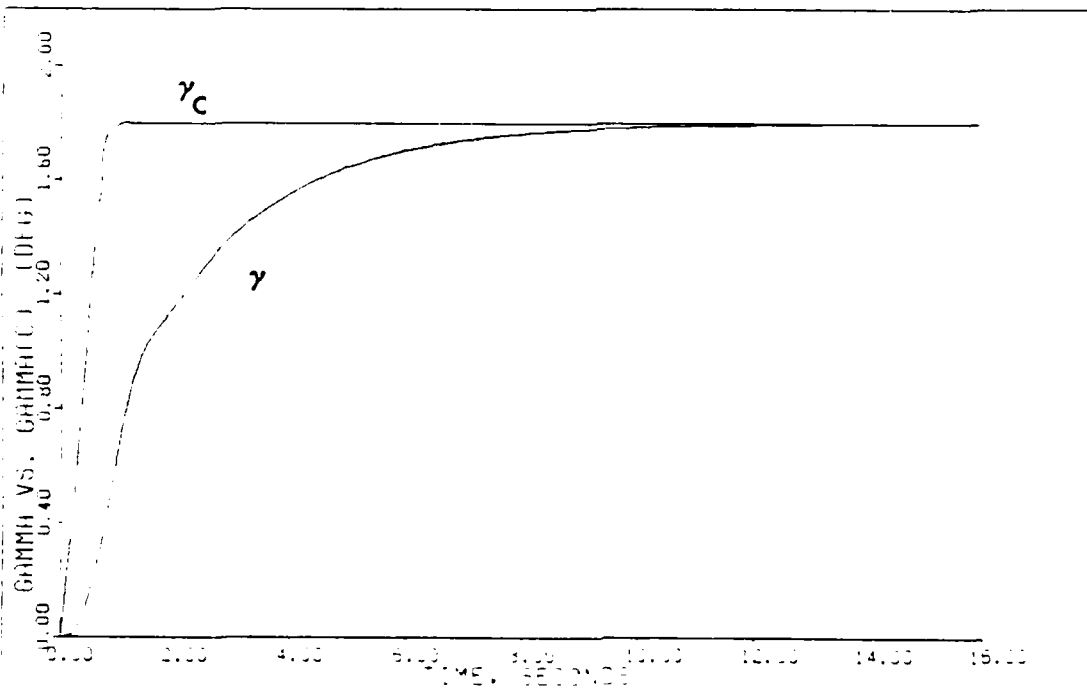
VERTICAL TRANS: -25% M-DELTA(H). (0.9M FL200)

Fig. 5.102



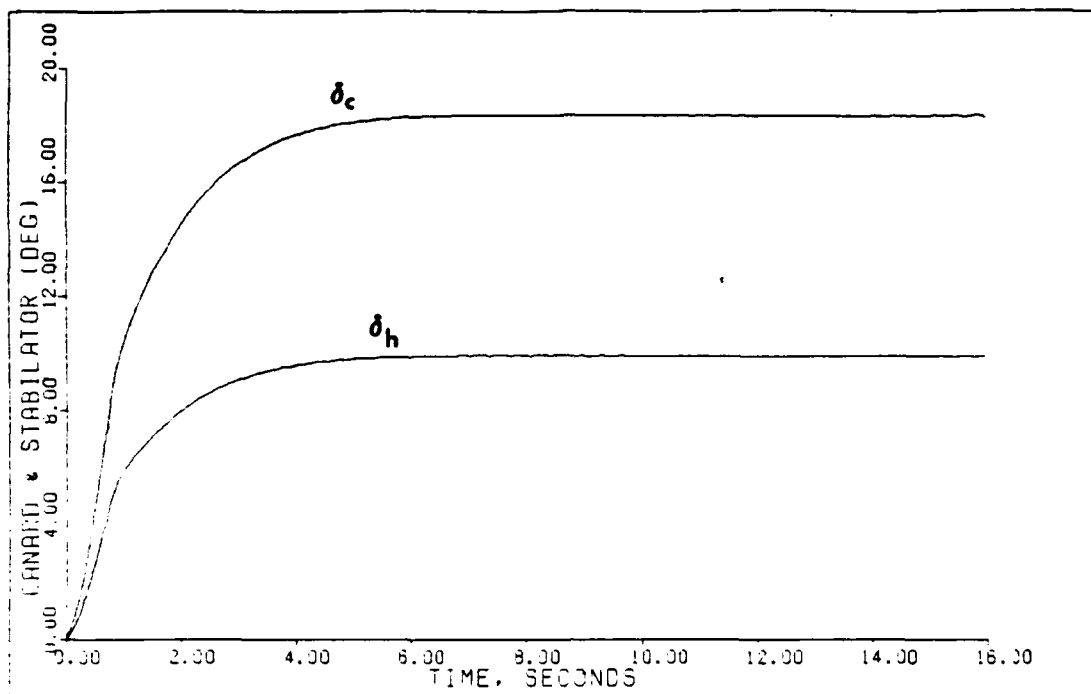
VERTICAL TRANS: -25% M-DELTA (H), 10.9M/FL200

Fig. 5.103



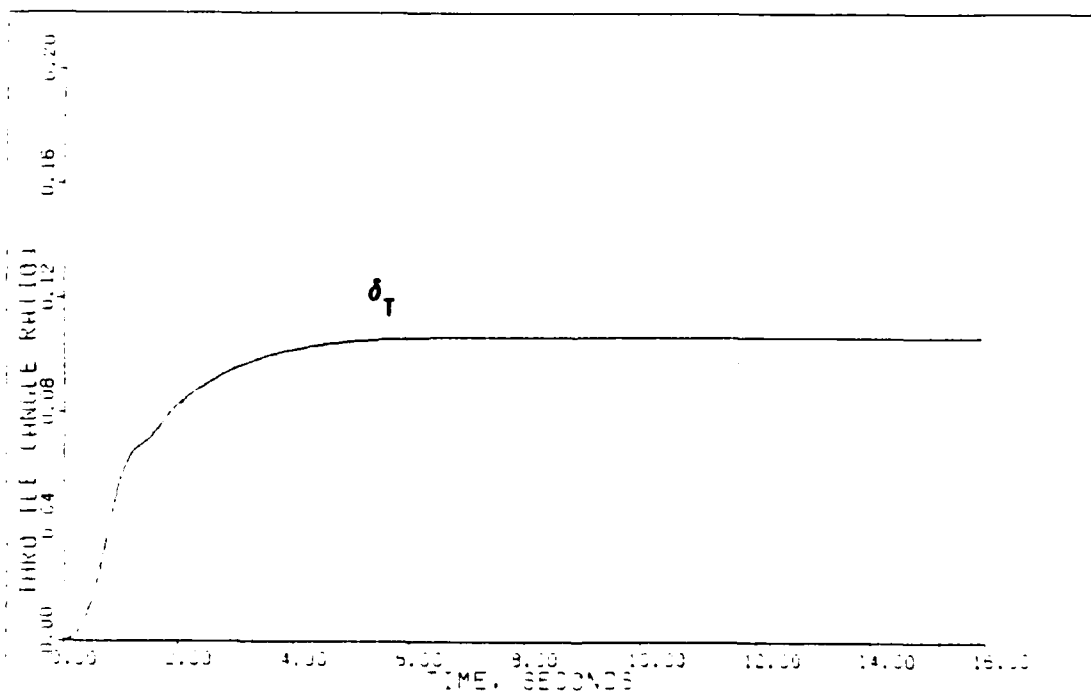
VERTICAL TRANS: -25% M-DELTA (H), 10.9M/FL200

Fig. 5.104



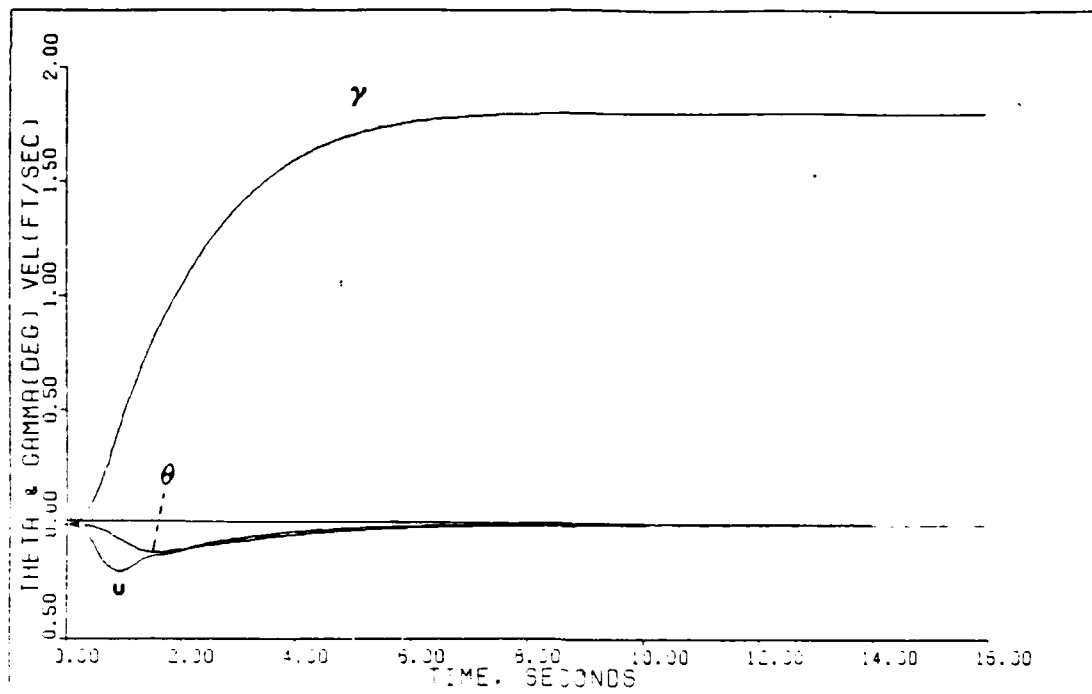
VERTICAL TRANS: +100% X-DELTA(H), (0.9M/FL200)

Fig. 5.105



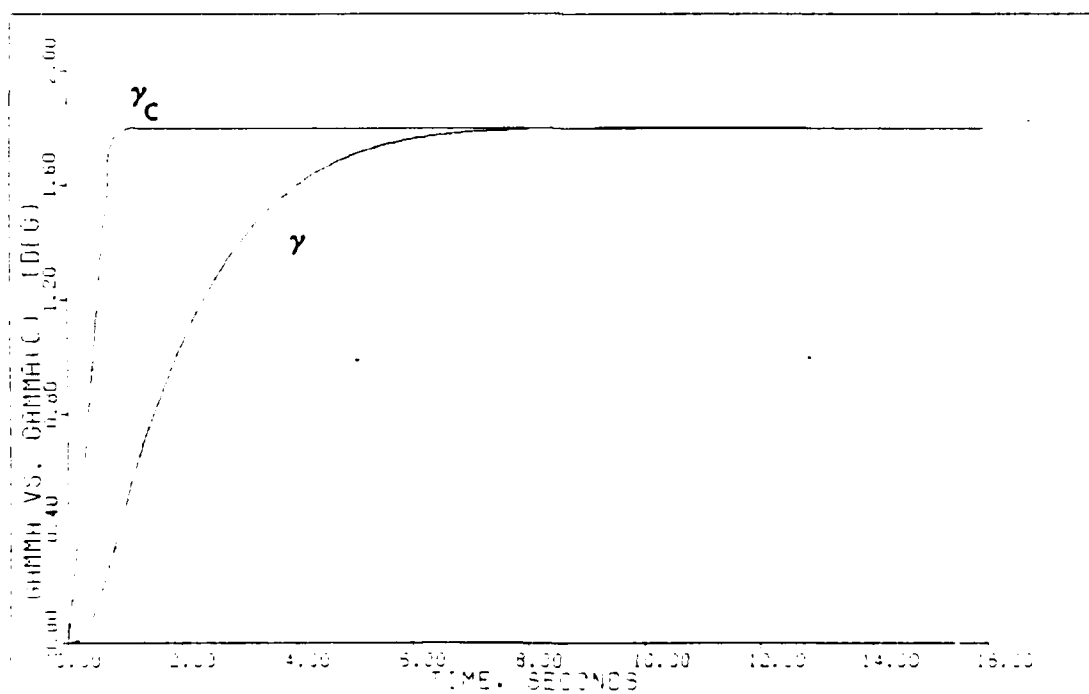
VERTICAL TRANS: +100% X-DELTA(H), (0.9M/FL200)

Fig. 5.106



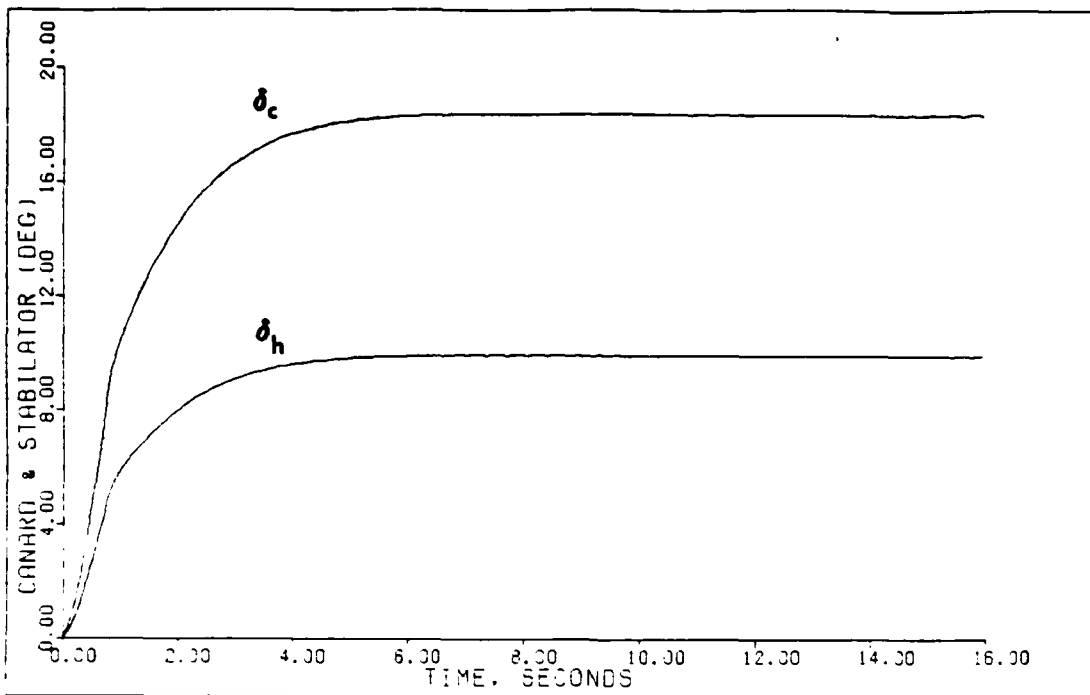
VERTICAL TRANS: +100% X-DELTA (H), 0.9M/FL200

Fig. 5.107



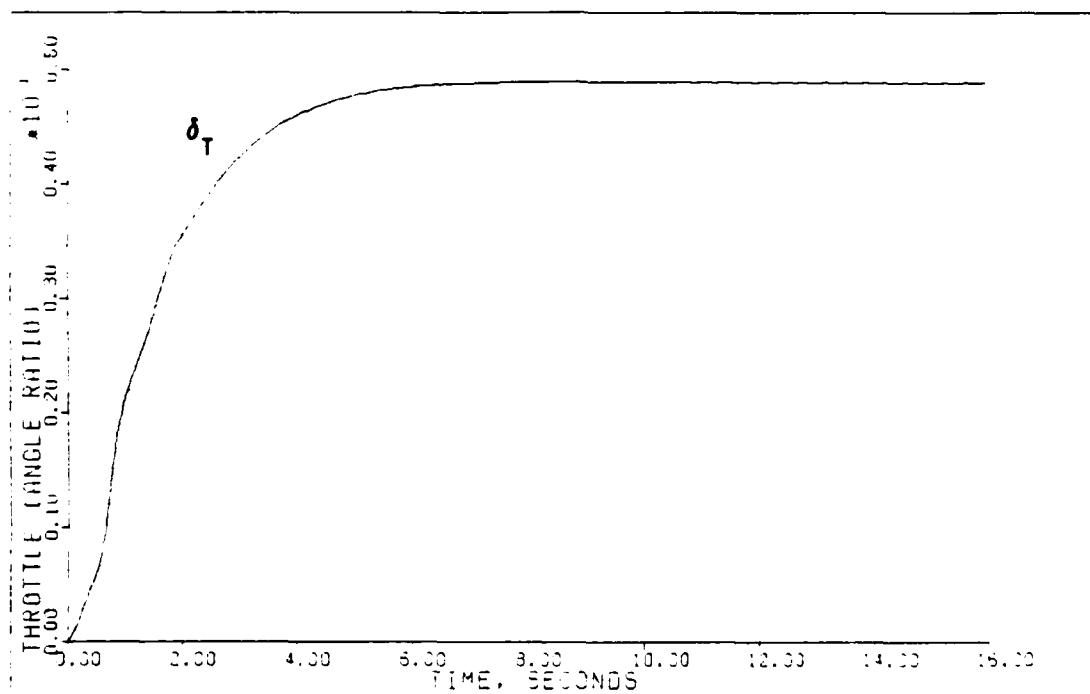
VERTICAL TRANS: +100% X-DELTA (H), 0.9M/FL200

Fig. 5.108



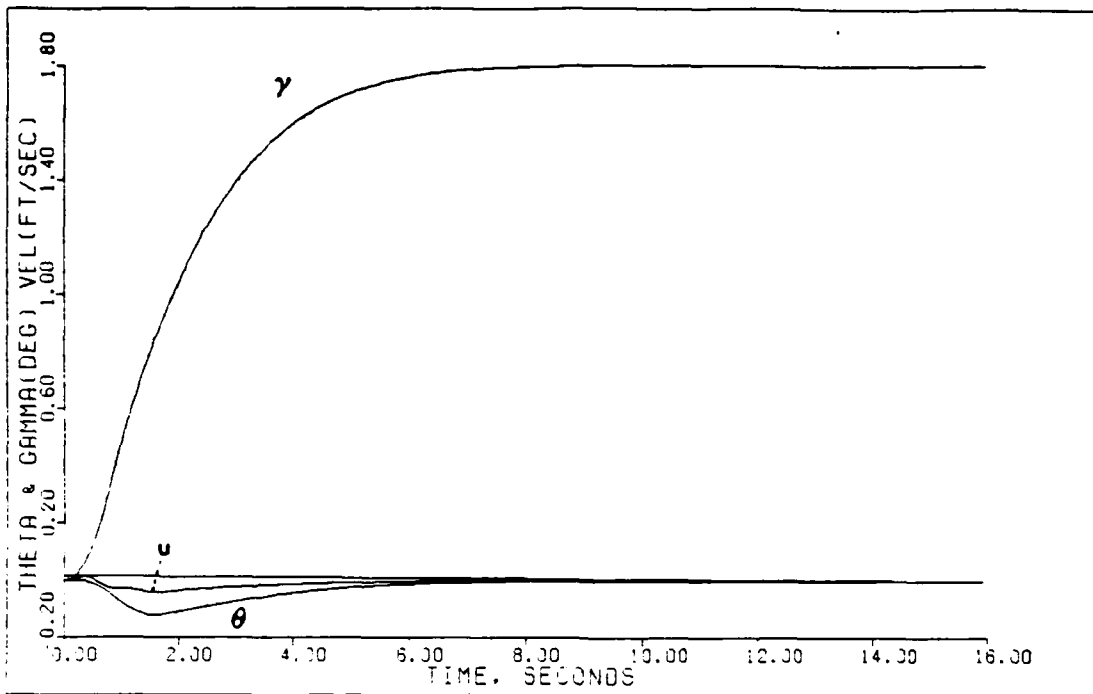
VERTICAL TRANS: -50% X-DELTA H, 10.9M/FL2000

Fig. 5.109



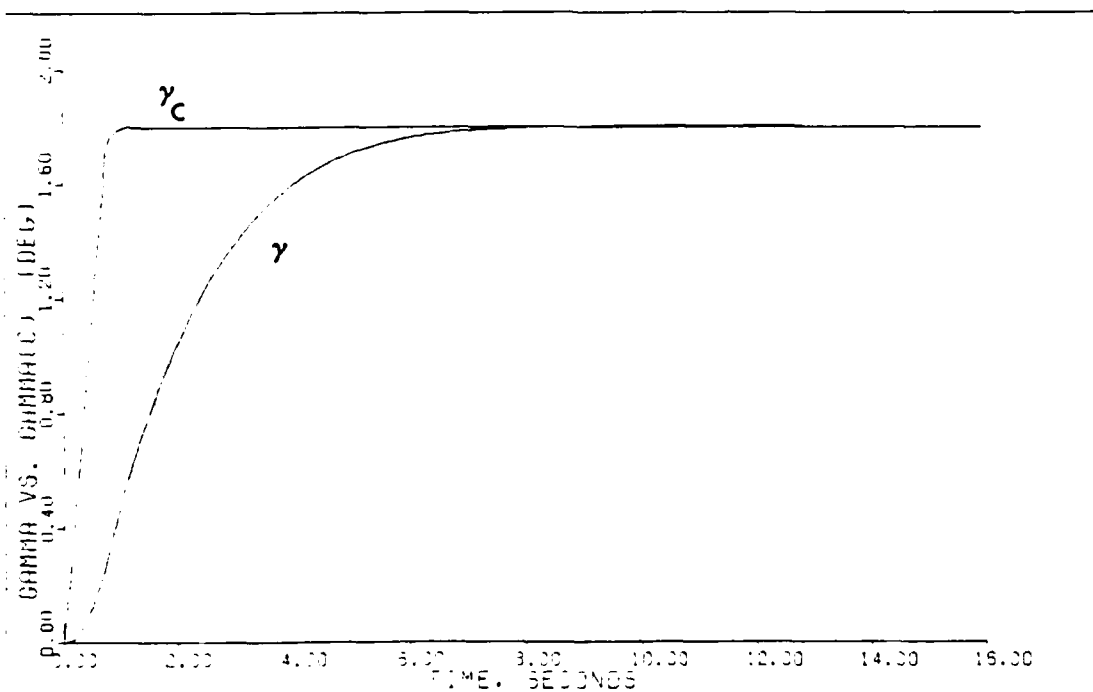
VERTICAL TRANS: -50% X-DELTA H, 10.9M FL2000

Fig. 5.110



VERTICAL TRANS: -60% X-DELTA (H) 0.9M FL200

Fig. 5.111



VERTICAL TRANS: -60% X-DELTA (H) 0.9M FL200

Fig. 5.112

The most insensitive control derivative tested is X_{δ_H} . The throttle input has no effect on the $\dot{\alpha}$ or the q equation; its only derivative is found in the \dot{u} equation. Consequently, when either the canard or stabilator drag coefficient is changed, the throttle compensates with thrust without affecting the q or $\dot{\alpha}$ equations. Figures 5.105 through 5.112 record the results of a +100 percent and -50 percent change in X_{δ_H} . As the stabilator creates more drag (+100 percent plots), the throttle advances to a higher value in steady-state. With less drag induced (-50 percent plots), the throttle settles at a lower power setting. None of these plots indicates any instability resulting from the variation in X_{δ_H} .

The previous results demonstrate an excellent amount of insensitivity to parameter variation.

5.7 Sensor Noise Results

The effects of sensor noise on system performance is demonstrated with the pitch pointing maneuver at 1.4 Mach/FL 200. Zero-mean, white gaussian noise is added to the outputs as they exit the sensor dynamics in the feedback path. Since this is an "irregular" Porter design, an additional state derivative, $\dot{\theta}$, is added to the feedback vector by means of the measurement matrix \underline{M} . The pitch rate ($q = \dot{\theta}$) must be sensed, however, before it can be fed back. As a consequence, noise enters the system through

this additional measurement. Appendix A explains in detail how MULTI is modified to simulate "noisy" measurements.

Chapter III points out that MULTI does not include sensor dynamics for the state derivative measurements but should be included in future improvements to the simulation program. The noise strength corresponds to the standard deviation of its distribution and is selected from a previous thesis which models rate and angle sensors used in the F-14 Tomcat (17). These typical standard deviations are given as:

$$\begin{aligned}\sigma_{\theta} &= 0.4760\text{E-}05 \text{ rads} \\ \sigma_{\alpha} &= 0.1220\text{E-}04 \text{ rads} \\ \sigma_u &= 0.5000\text{E-}04 \text{ ft/sec} \\ \sigma_{\gamma} &= 0.1310\text{E-}04 \text{ rads} \\ \sigma_q &= 0.3220\text{E-}04 \text{ rads/sec}\end{aligned}$$

The value for velocity was picked arbitrarily since no typical values were available in the literature. In addition, since gamma is the difference of the two measured angles theta and alpha, its standard deviation is computed as:

$$\sigma_{\gamma} = \sqrt{\sigma_{\theta}^2 + \sigma_{\alpha}^2} \quad (5-3)$$

The sensor noise simulation is conducted using the fully developed four-state model (actuators, delay, and sensors) with no control limits installed. The control

limits are eliminated to prevent instability which occurs when the control inputs reach their stops. Instability resulting from control input saturation is characteristic of integral controllers since the controller output signal is proportional to the integral of the error from the initial time to the present. The phenomenon is known as "wind-up" error. This is why control input saturation is avoided in all of the designs accomplished in this study.

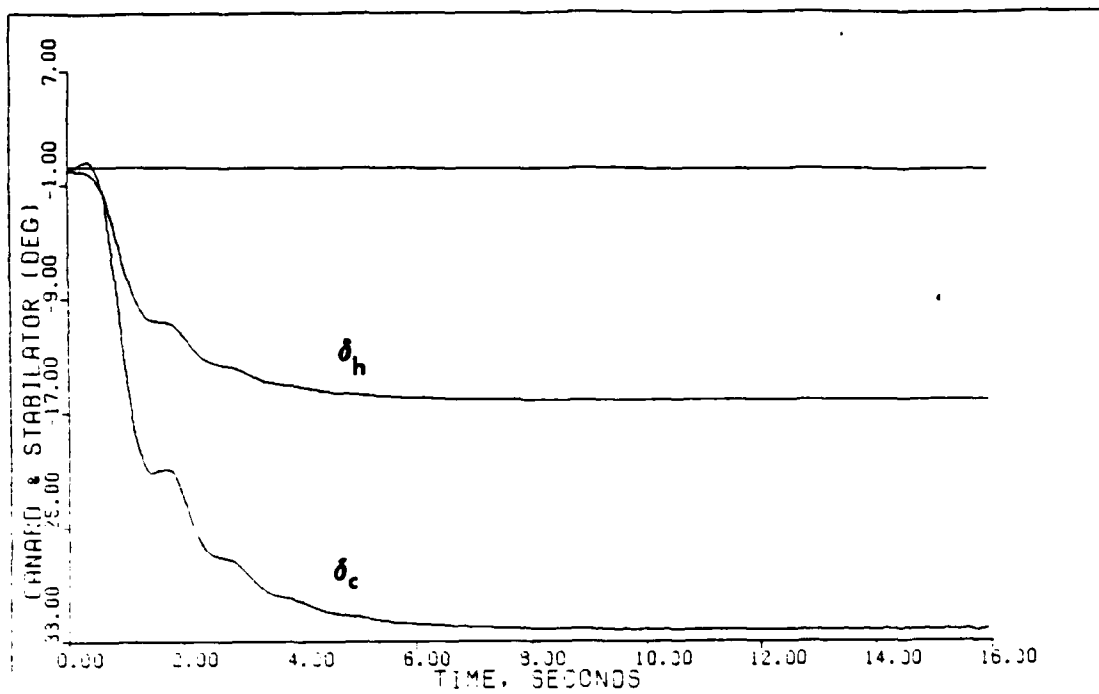
In order to achieve statistical significance, the noise simulation results are the average of five independent simulations, each using different random noise vectors with the same standard deviation. The additional option of plotting the standard deviation of the errors over the simulation ensemble is not available in the current modification to MULTI. This feature is suggested for follow-on research.

Figures 5.113 through 5.120 present the sensor noise effects on the control inputs. Figures 5.113 and 5.114 show the responses with no noise and are included for comparison purposes. Figures 5.115 and 5.116 show results using the typical noise levels presented above. There is a mild oscillation present in all of the control inputs but is largest in the canard. This result is consistent with the results from parameter variation instability, i.e. the canard is always affected most in the presence of dynamic instability.

The frequency of oscillation is distorted by the plotting routine since only a fraction of the total data points are used for the plot. The actual data has a frequency of 20 Hz (half the sampling frequency) while the plot indicates 2.95 Hz. In addition, the plot data is the output from the controller before it is affected by actuator dynamics. Given the frequency response of the actuators at 20 Hz (Chapter III), the control input deflections would be attenuated by approximately 12.5 dB or 76 percent.

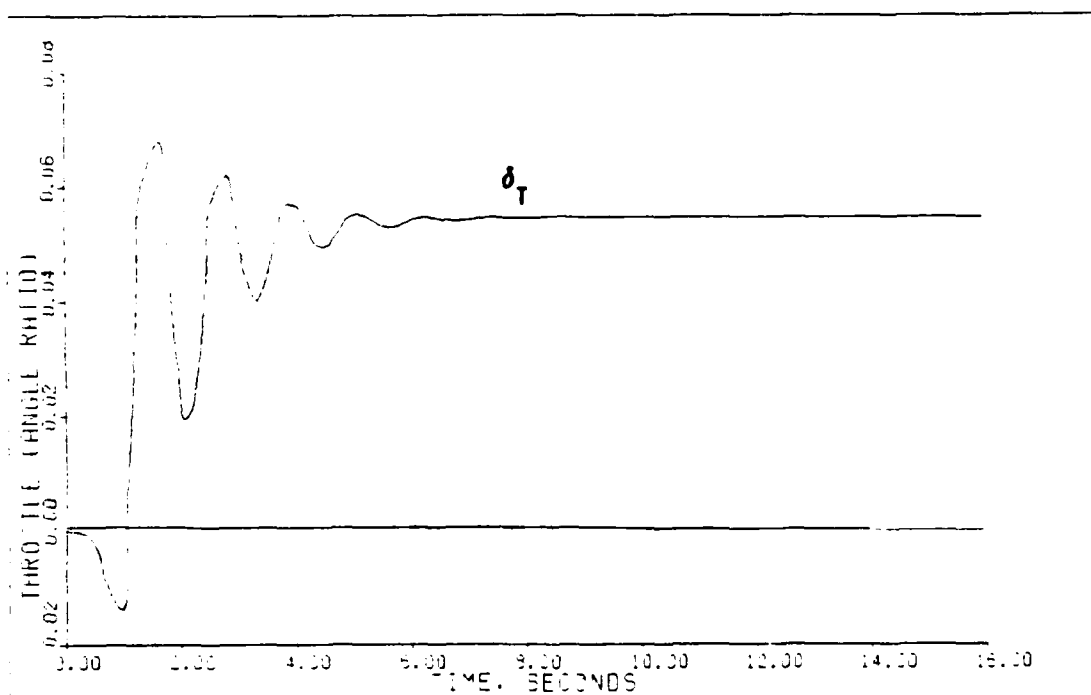
Figures 5.117 and 5.118 show the effects of 2.5 times the previous noise levels (2.5x). The oscillation is now greater in amplitude but still stable and centered about the steady-state control input deflection value. As the noise level is increased to 5 times the typical level (5x), divergence appears in the throttle input (Figures 5.119 and 5.120). The canard's oscillation amplitude causes a plotting distortion that appears as a surface deflection that moves backward in time. The only piece of valuable information gained from Figure 5.119 is that the control oscillations remain centered on their "noise-free" steady-state values.

The throttle, however, exhibits a characteristic known as "random walk" (14). Since unfiltered, white gaussian noise is being fed back and integrated by the PI controller, the observed result is Brownian motion (random



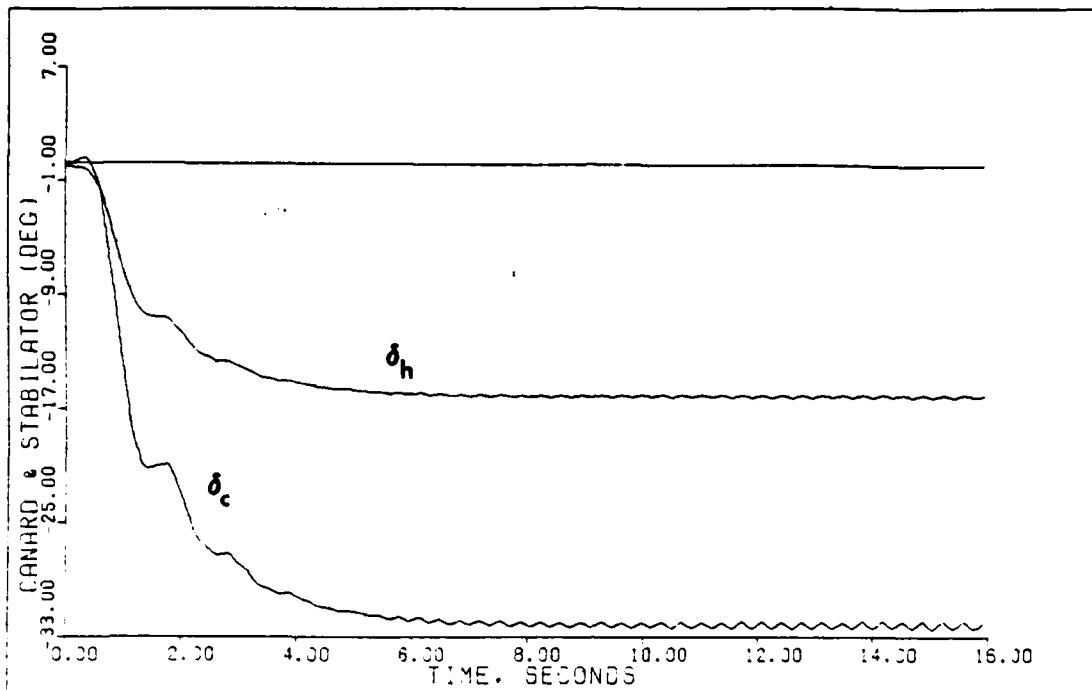
PITCH POINTING: NO SENSOR NOISE (1.4M/FL200)

Fig. 5.113

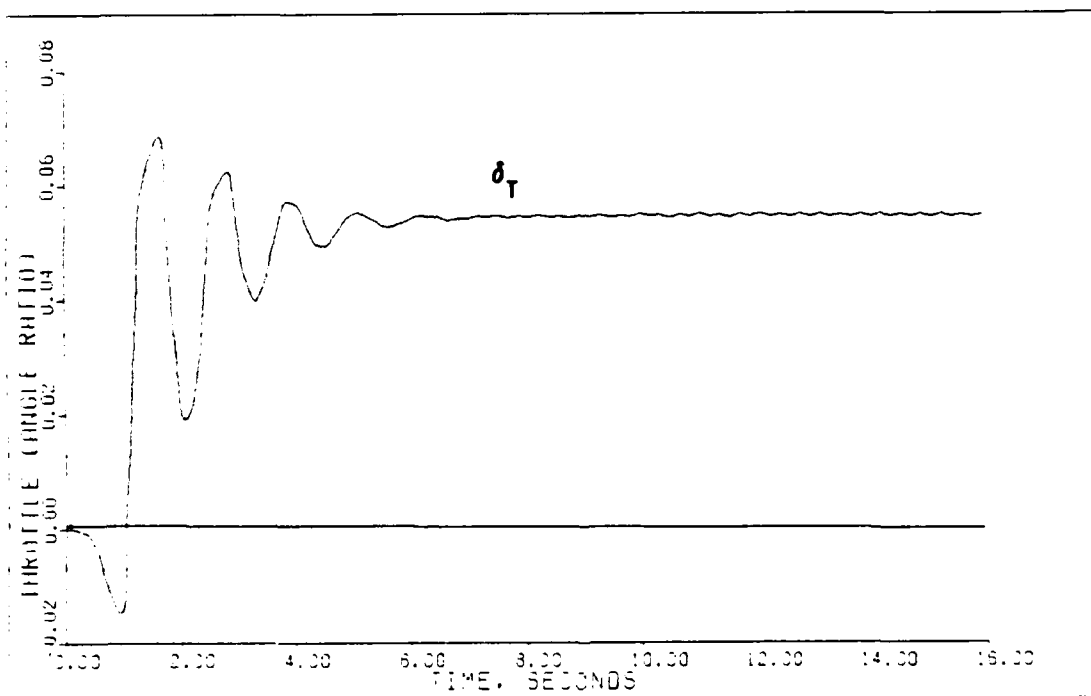


PITCH POINTING: NO SENSOR NOISE (1.4M/FL200)

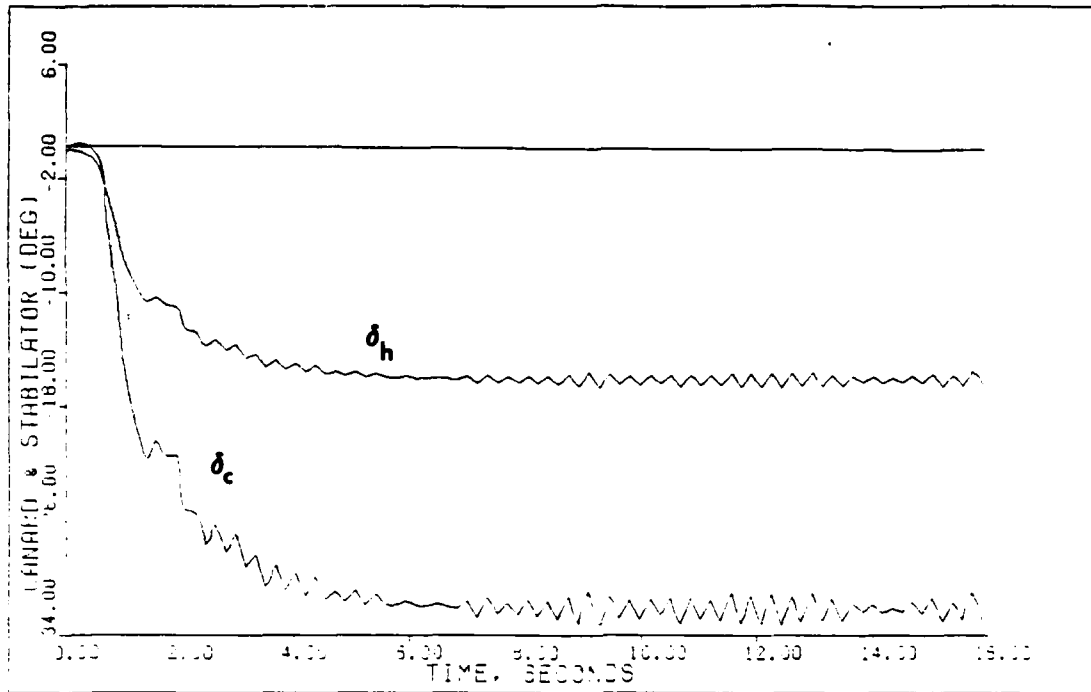
Fig. 5.114



PITCH POINTING: TYPICAL SENSOR NOISE (1.4M FL200) Fig. 5.115

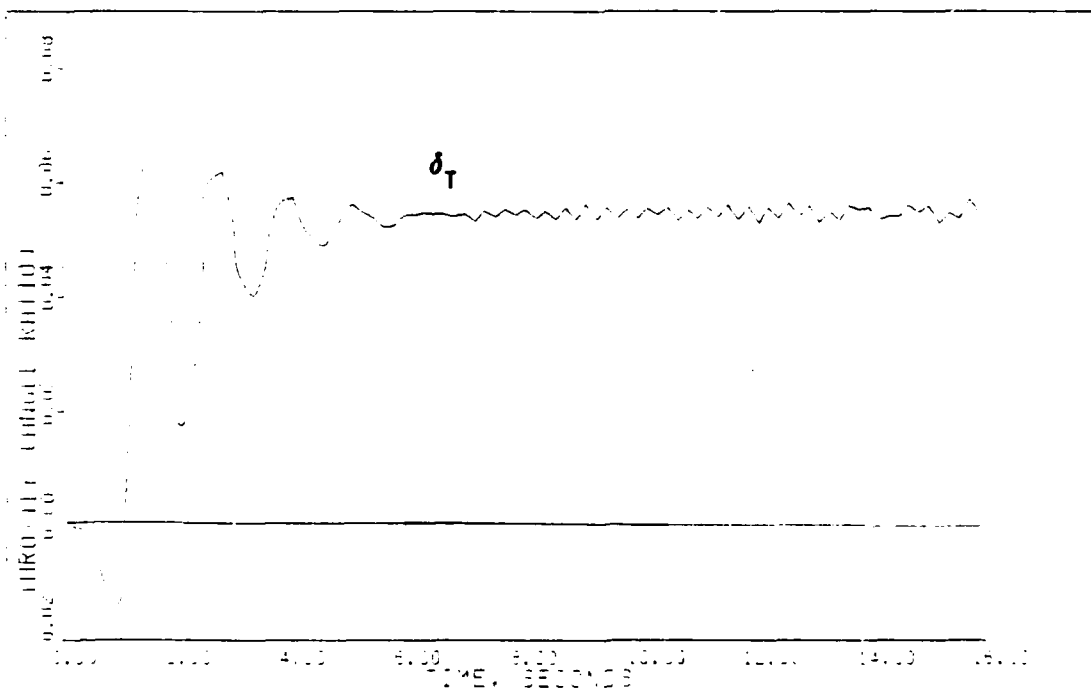


PITCH POINTING: TYPICAL SENSOR NOISE (1.4M FL200) Fig. 5.116



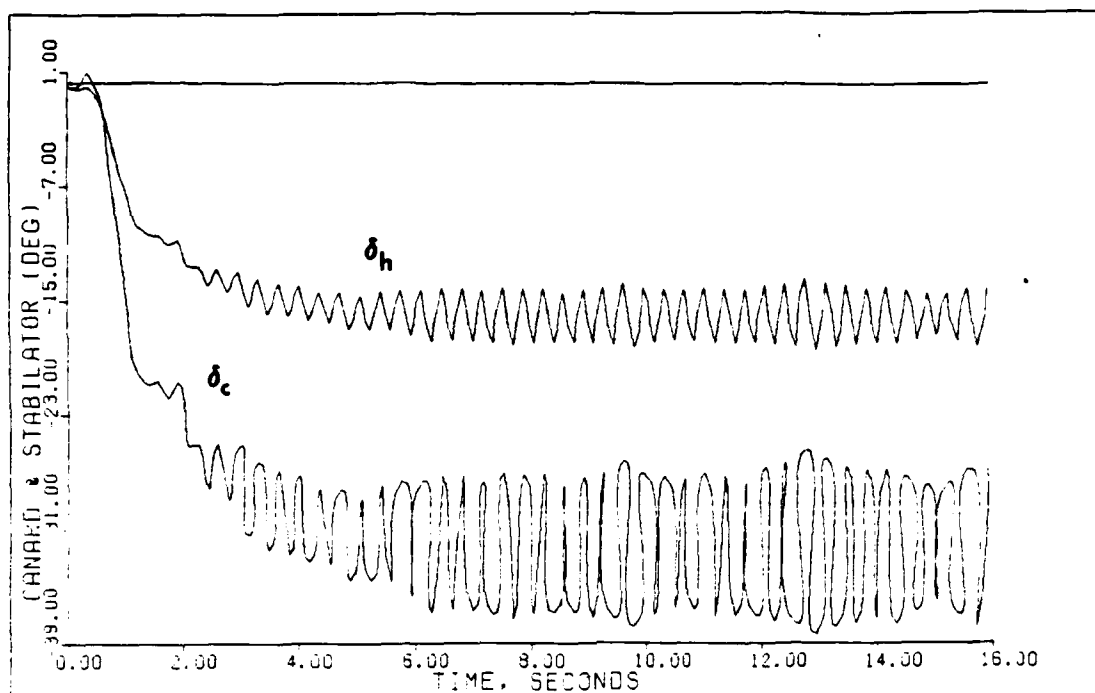
PITCH POINTING: 2.5K TYPICAL SENSOR NOISE (1.4M/FREQ)

Fig. 5.117



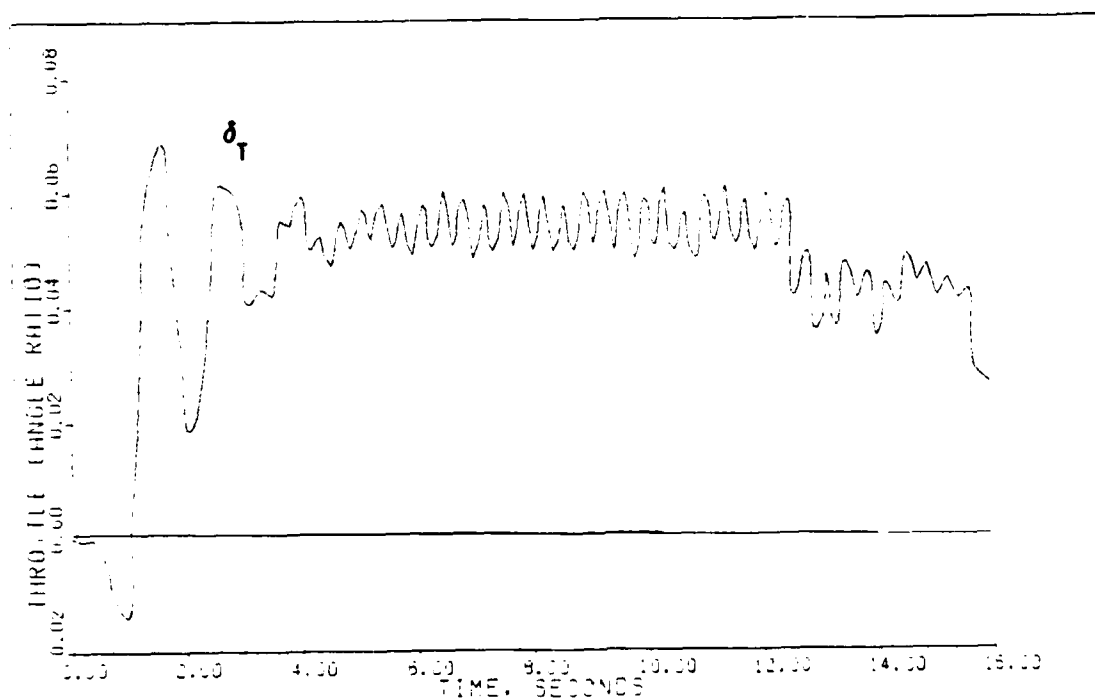
PITCH POINTING: 2.5K TYPICAL SENSOR NOISE (1.4M/FREQ)

Fig. 5.118



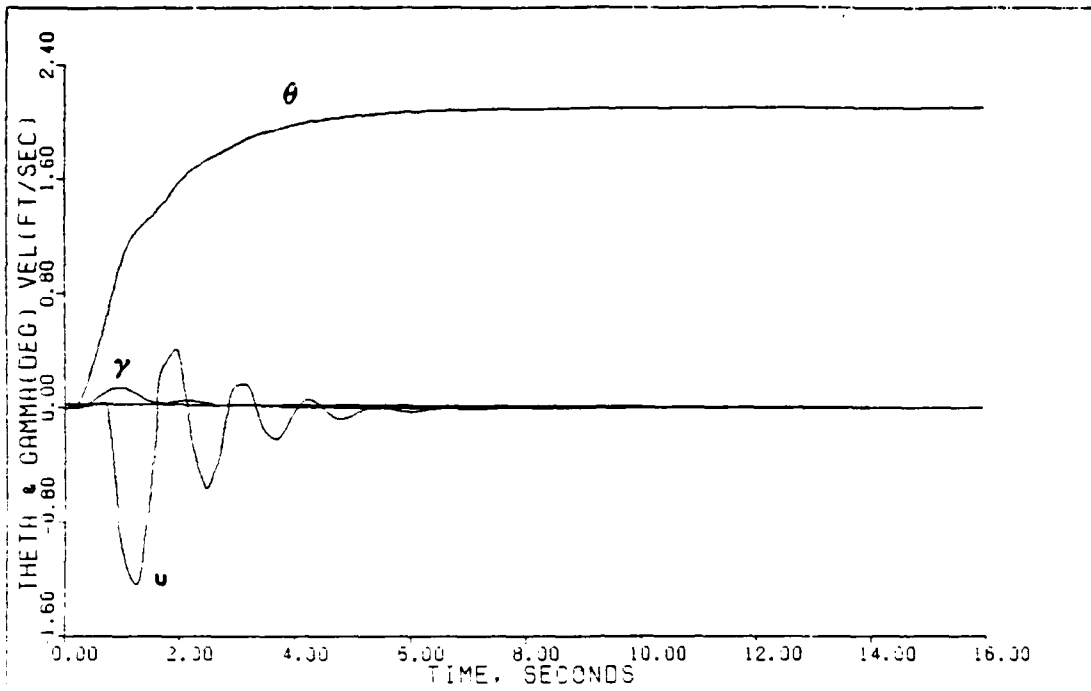
PITCH POINTING: 5X TYPICAL SENSOR NOISE (1.4M/FL200)

Fig. 5.119



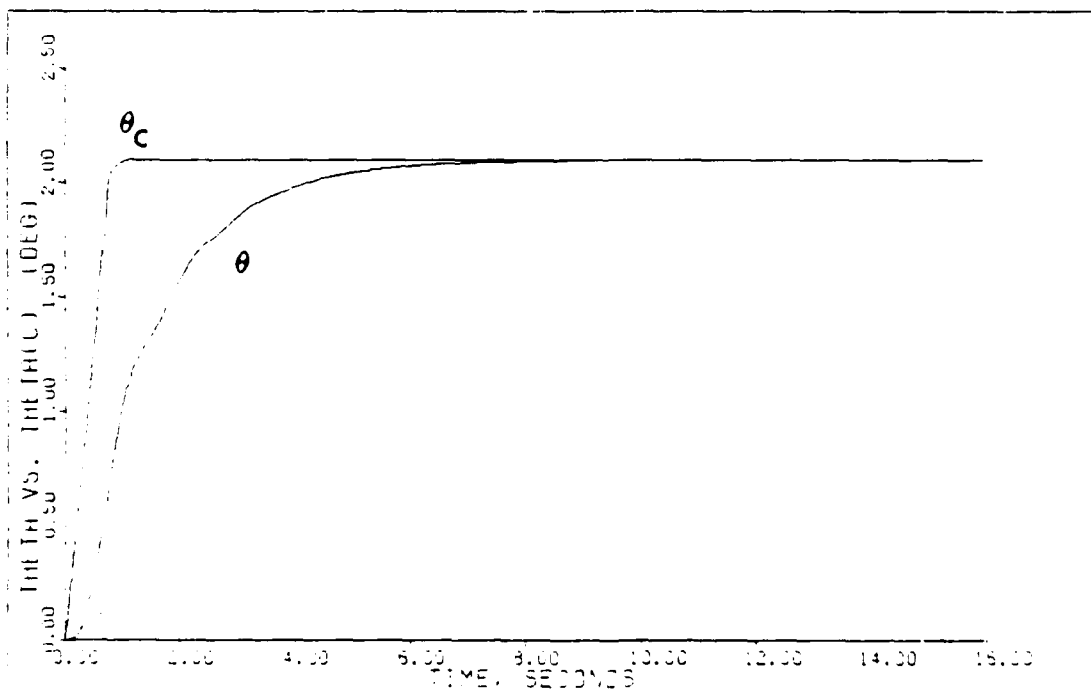
PITCH POINTING: 5X TYPICAL SENSOR NOISE (1.4M/FL200)

Fig. 5.120



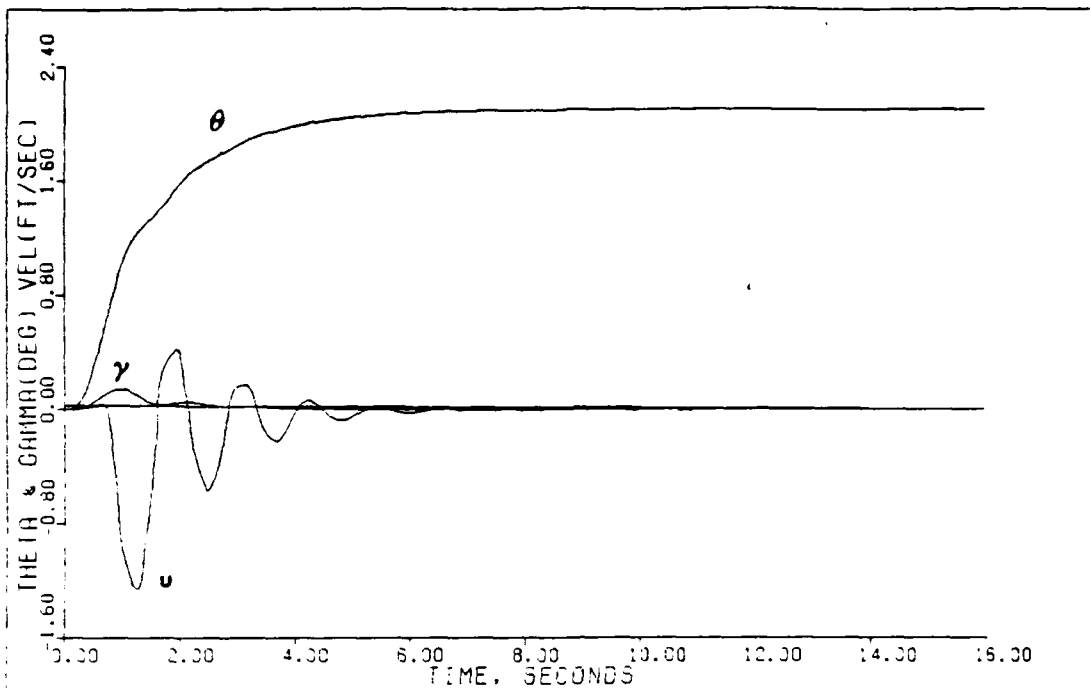
PITCH POINTING: NO SENSOR NOISE (1.4M/FL200)

Fig. 5.121

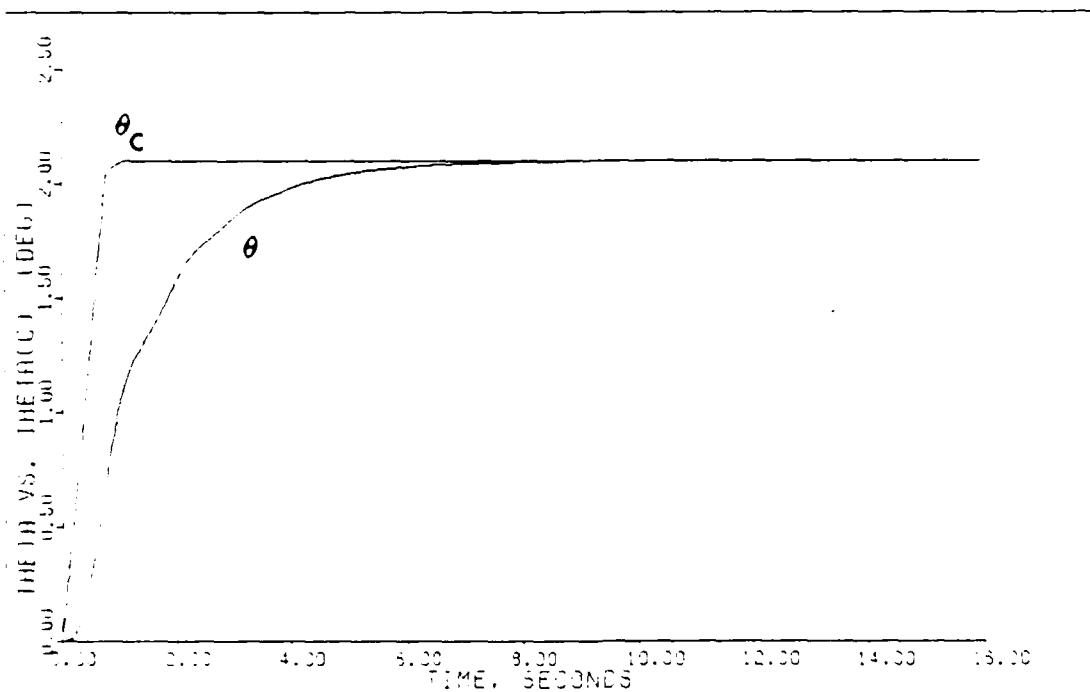


PITCH POINTING: NO SENSOR NOISE (1.4M/FL200)

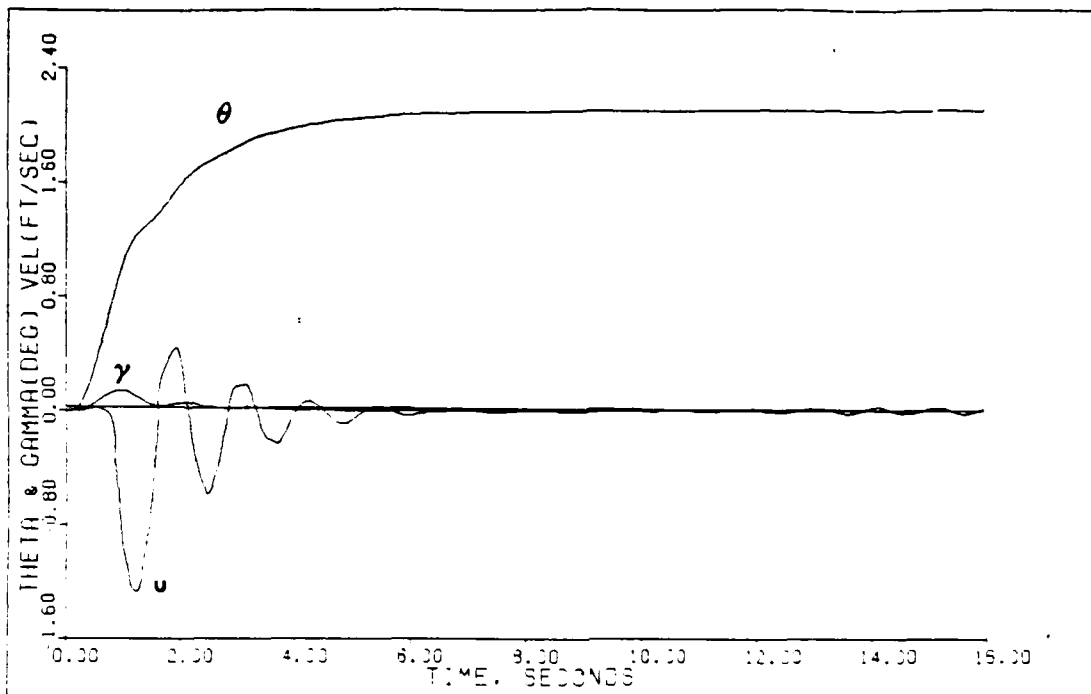
Fig. 5.122



PITCH POINTING: TYPICAL SENSOR NOISE 11.4M FL200 Fig. 5.123

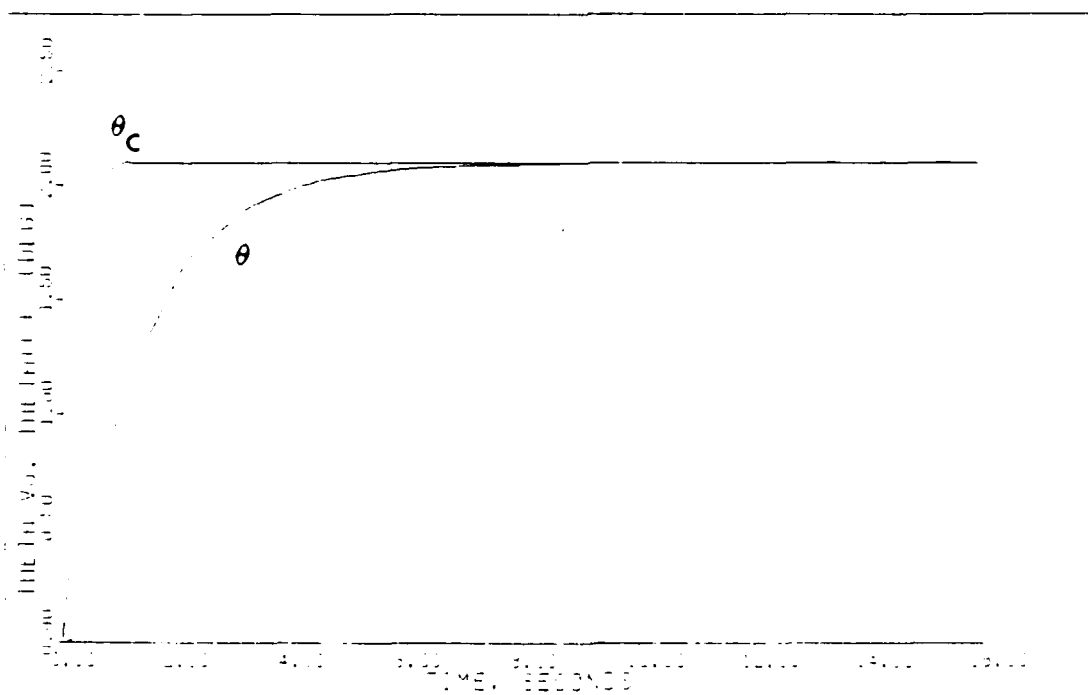


PITCH POINTING: TYPICAL SENSOR NOISE 11.4M FL200 Fig. 5.124



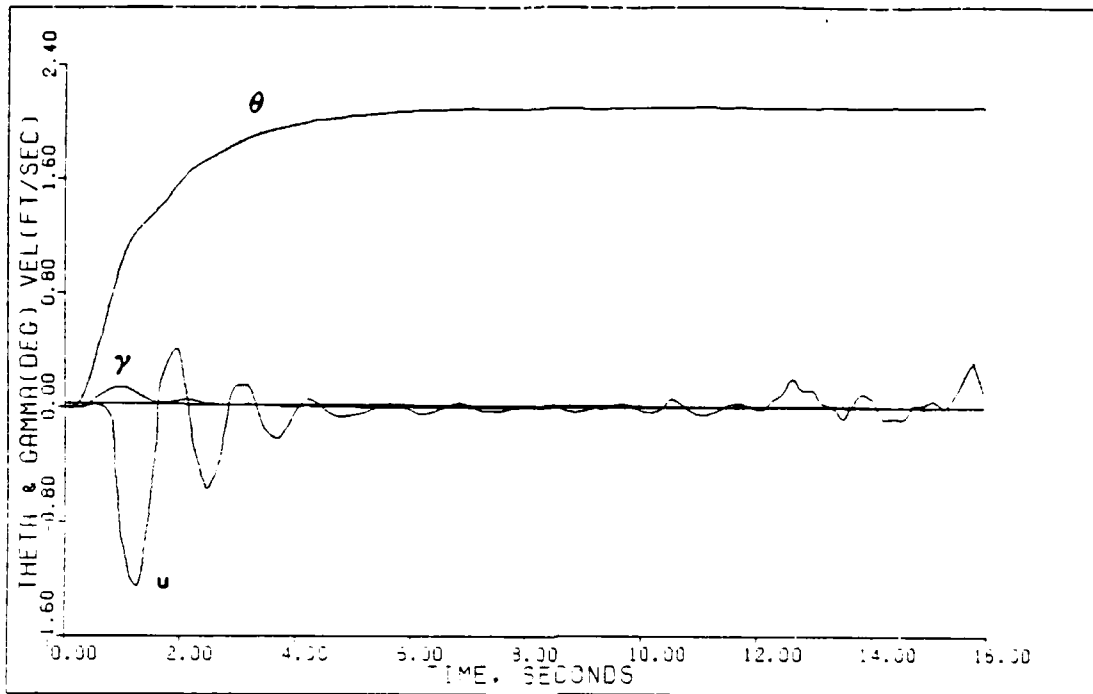
PITCH POINTING: 2.5K TYPICAL SENSOR NOISE (1.4M F1000)

Fig. 5.125



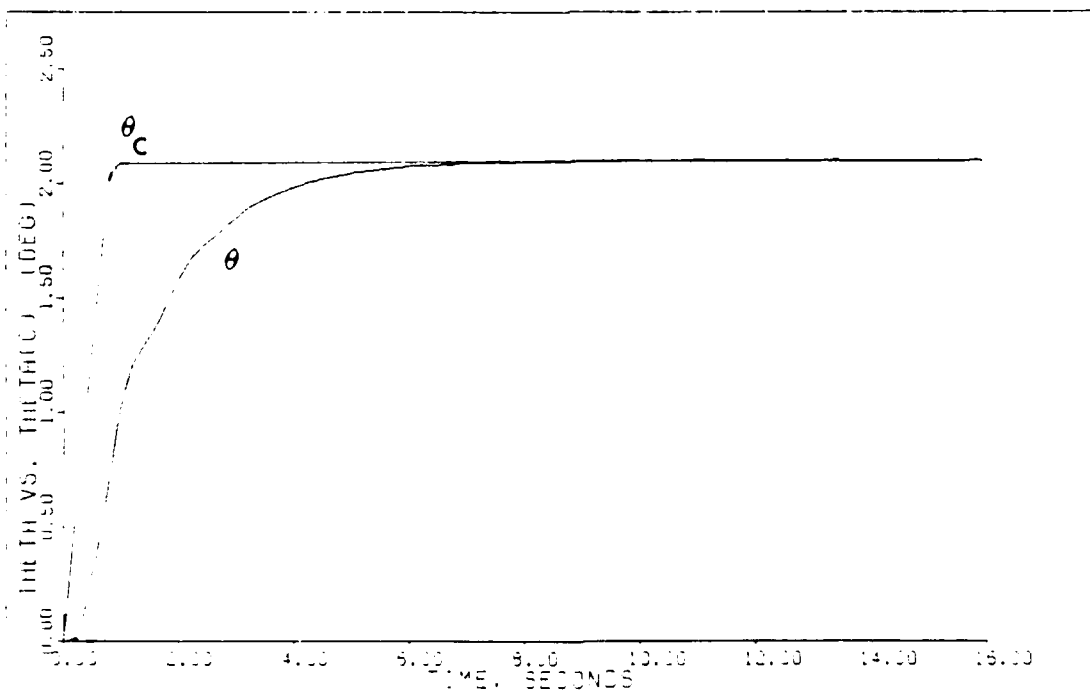
PITCH POINTING: 2.5K TYPICAL SENSOR NOISE (1.4M F1000)

Fig. 5.126



PITCH POINTING: 5X TYPICAL SENSOR NOISE (1.4M/FLOCC)

Fig. 5.127



PITCH POINTING: 5X TYPICAL SENSOR NOISE (1.4M/FLOCC)

Fig. 5.128

walk) in the throttle input. This effect can be reduced by the addition of a noise filter in the feedback path of the model. The result is a system that can tolerate higher noise levels without exhibiting instability.

Figures 5.121 through 5.128 compare the output responses as the sensor noise level is increased. There is virtually no apparent effect in either theta or gamma since the canard and stabilator do not diverge, even at the highest noise level. An insignificant oscillation is seen in velocity at the 2.5x noise level (Figure 5.125). Because of the throttle divergence at the highest noise level (5x), the perturbation velocity begins to exhibit instability beyond the 12 sec point in the simulation (Figure 5.127). The velocity instability is coincident with the point of throttle divergence, as expected.

Overall, the system demonstrates an excellent insensitivity to typical levels of sensor noise. The addition of a noise filter would further improve these results and is recommended for future work.

5.8 Summary

The results presented in this chapter demonstrate the outstanding capability and design flexibility of a "Porter configured," proportional plus integral controller. Several maneuvers are performed over a wide range of flight conditions with the results indicating a high degree of

output decoupling and system stability. The responses with the basic model are smooth and well behaved, characteristic of an ideal, high gain, error-actuated control system. Even in the case of fully developed models (all dynamics included), the responses are stable and well behaved.

In addition, the effects of parameter variation and sensor noise are investigated with great success. The control laws are shown to be robust under conditions of control derivative variation and virtually unaffected by typical values of sensor noise.

Chapter VI presents the conclusions drawn from the results presented in this chapter, along with recommendations for future work.

VI. Conclusions and Recommendations

6.1 Design Results

The Porter design method is proven to be a powerful and straightforward technique for the design of multi-variable control laws for the F-15/STOL derivative fighter. The previous chapters demonstrate the capability of achieving at least satisfactory and, in most cases, excellent design results throughout the flight envelope. These smooth and stable maneuvers are accomplished with a model that exhibits open-loop static instability for a wide range of dynamic pressures.

All of the maneuvers are performed while adhering to the realistic limitations of control input rate and deflection limits. The deflection limits are a built-in part of MULTI's simulation; however, rate limits must be determined from the output responses. The lack of internal rate limits within the simulation can present a problem and is discussed later in this chapter. In addition to input limitations, all of the control laws demonstrate a high degree of output decoupling, even when using the fully developed model with all delays and dynamics added.

There is a note of caution that must be pointed out when interpreting the results of this report. Strict

attention is paid to the conditions of linearity established in Chapter III for the aircraft output angles and rates. Control input linearity is no longer valid, however, when the full range of control deflection is commanded during the simulation.

A more accurate approach to this problem would be made by the use of a nonlinear simulation using stored, wind tunnel aero data defined for each of the control inputs. This simulation capability is not available for this study. The goal of this research, however, is the validation of a design technique and to this end, the goal is achieved.

The Porter design techniques are shown to be extremely flexible and easily used in achieving a satisfactory design. As pointed out in Chapter IV, the parameters of σ and $\bar{\alpha}$ provide direct insight into the closed-loop root migration of the resulting design. This insight helps the designer "tune" the controller for a specific response in minimum time.

6.2 Design Process Improvements

One of the most severe limitations with this design process is the requirement for independence of all control inputs. Any aircraft model has only six degrees of freedom since it takes six equations (3 moment and

3 force) to completely describe its three-dimensional motion. Thus, only six independent inputs are allowed.

This thesis studies the longitudinal motion of the F-15/STOL which is represented by three of these six equations. Advanced aircraft designs using redundant control surfaces generally have more inputs available than this design method can allow. The F-15/STOL has five independent longitudinal inputs: canard, aileron, flaperon, stabilator, and nozzle. The inputs can be combined into a single input as is done in the constant g pull-up maneuver for which the three-state model is used; however, design flexibility is reduced in this case. The flexibility is limited since both inputs must now operate simultaneously and always in the same direction (or opposite each other) if combined in this manner. In addition, a rather large simplification must be made with the combined actuator dynamics model.

A solution to this problem is the introduction of a control input weighting function that eliminates the independence between the redundant surfaces. This weighting function allows for the dependent operation of an increased number of control inputs that all operate with their own actuator dynamics. Mathematically, there can never be more than three independent inputs in a three degree of freedom model if a unique solution is to be determined. The addition of a weighting function, however,

allows for more potential inputs to be part of the three input limit.

An additional benefit from this improvement is the ability to tailor the peak transients of the control inputs through the coefficients of the weighting function. Frequently, a control input will saturate during the transient period of a maneuver while other, equally capable inputs, are not fully used. This condition generally establishes an upper bound on the response time and maximum maneuver capability. If the weighting on the inputs could be adjusted by the designer, the maneuver could be essentially optimized by the input responses.

6.3 MULTI Improvements

From the experience gained during the course of this research, several improvements to the program, MULTI, are desirable and suggested for future studies in order to enhance the design.

First, the noise simulation capability added as a result of this study and a similar thesis by Acker (1), needs further improvement. An option that allows for noise filter dynamics within the simulation is necessary. The random walk effects described in Chapter V are the result of integrating unfiltered, white gaussian noise. These filters can be a normal part of control systems and therefore should be included in the design process.

In addition, a complete Monte Carlo analysis should include a calculation (and plot) of the 1-sigma values of the simulation ensemble. For example, if the Monte Carlo analysis consists of five simulations using a different random noise vector for each run, the average values over the simulation time are now computed and plotted by the current version of MULTI. For statistical significance, however, each individual run should be compared against the average, the differences squared, summed, and the square root taken of the total. This process is computed at each sample time and plotted as a 1-sigma function. The convergence of this result would determine the validity of the noise analysis.

Second, sensor dynamics must include the inner-loop compensation measurements determined by the M matrix in irregular designs. The current option provides for dynamics only on the output variables.

Third, rate limits for all control inputs need to be incorporated within the simulation. This provision would eliminate the high frequency content of the input responses and provide for a more accurate simulation.

Fourth, every design requires that the conditions of controllability and observability be met. The program ZERO calculates these conditions by listing the invariant zeros of the system matrices. Currently, any design attempt is a two-step process, first using ZERO to establish the

acceptability of the output scheme; and second, re-entering all of the system matrices into MULTI to begin the design. Incorporation of ZERO into MULTI would be a convenient addition.

Fifth, the designer needs to see all of the closed-loop roots of the system prior to simulation. Currently, option 6 is capable of presenting the closed-loop roots resulting from the system matrices and the integrator states. Since actuator and sensor dynamics are put in cascade with the plant, obtaining the eigenvalues of the complete closed-loop system is desirable for stability analysis.

Sixth, plots of the u vector are currently the inputs to the actuators. These control input plots represent the output of the PI controller. Provision should be made for plotting the responses of the actual inputs (canard, stabilator, etc.) deflections, taking into account the actuator dynamics.

Seventh, the second step for improving the non-linear simulation capability of MULTI is made by the solution to the "sign swapping" problem encountered in this thesis. The first was the addition of input deflection limits. A further enhancement would be the capability to change the B matrix entries between sample periods to better account for the nonlinear effects of full deflection inputs. This method is identical to the "look-up table" method discussed earlier in the chapter.

Eighth, the theory of the Porter method allows for the designer to completely specify the asymptotic root migration of the closed-loop plant by selection of the $\bar{\alpha}$ and sigma design parameters (regular design). The details of this concept are described in Appendix B. The locations of the finite roots are determined by the relationship between \underline{K}_0 and \underline{K}_1 .

MULTI reduces the designer's flexibility somewhat by using $\bar{\alpha}$ as a simple proportionality constant between these two matrices. As a result, all the finite roots resulting from the integrator states are placed at the same location, $s = -\bar{\alpha}$. As shown in Chapter IV, these roots often play a dominant role in the time response of the system. Therefore, flexibility in the placement of these roots is sometimes desirable. By replacing $\bar{\alpha}$ with a diagonal matrix similar to sigma, the designer is free to choose the placement of these roots without having to mutually change all of the elements of the \underline{K}_1 matrix, the only method currently available.

Ninth, option 28, figures of merit, needs to be expanded to include the responses of states, control inputs, or combinations thereof. The responses of interest are not limited to only those of the output vector y .

6.4 Future Research

The original plan for this research effort included the modeling and simulation of thrust vectoring. The requirements for the simulation of a variable nozzle input have been completed in this study and a similar thesis by Acker (1). The problem that remains is the determination of a proper input vector that will achieve a steady-state solution. Flight path is proportional to excess thrust which is controlled, in part, by the throttle input. The z-direction force and y-axis pitching moment, however, are both a function of the throttle input and the nozzle deflection angle. As a consequence, the input vector must be determined by the solution to a set of four nonlinear equations used in the simulation. Future research should address this problem.

Despite this setback, the results from this research using a fixed-nozzle model forms a baseline that future studies can use for comparison and analysis.

Index for Appendix A

	Page
Introduction	204
List of Changes and Additions	204
A. Gaussian Noise Option	204
1. Description	204
2. User's Guide	205
3. Programmer's Guide	211
B. Custom Input Option	219
1. Description	219
2. User's Guide	219
3. Programmer's Guide	224
C. Suppression of Actuators and Sensors	238
1. Description	238
2. User's Guide	238
3. Programmer's Guide	239
D. Saving Memory Files Without Exit	240
1. Description	240
2. User's Guide	240
3. Programmer's Guide	241
E. Convert Input Vector "U" From Radians to Degrees	242
1. Description	242
2. User's Guide	243
F. Plot Combination of States and Inputs	243
1. Description	243
2. User's Guide	243
3. Programmer's Guide	246

	Page
G. Simulation of Nonlinearities Peculiar to Aircraft	248
1. Description	248
2. User's Guide	249
3. Programmer's Guide	250
H. Calculate Initial Integrator State Vector Z0	253
1. Description	253
2. User's Guide	254
3. Programmer's Guide	255
I. Program Outline	256
1. Introduction	256
2. List and Description of Major Program Elements	256
3. Memory Files	261

Appendix A: Additions to MULTI

Introduction

During the course of this thesis, a number of changes and additions were made to the computer program MULTI to facilitate current and future research efforts. This appendix describes each of these changes and additions, as well as providing an outline of the entire program for the benefit of future programmers.

List of Changes and Additions

- A. Gaussian noise option
- B. Custom input option
- C. Suppression of actuators and sensors
- D. Saving memory files without exit
- E. Convert input vector "u" from radians to degrees
- F. Plot combination of states and inputs
- G. Simulation of nonlinearities peculiar to aircraft
- H. Calculate initial integrator state $Z(0)$ vector
- I. Program outline

A. Gaussian Noise Option

1. Description. This addition gives the user the option of simulating zero and non-zero mean, white,

gaussian, noise inputs to the system during execution of option 26. There are three types of noise inputs available, distinguished by the place in which the noise is injected into the linearized model. The first type, output measurement noise, is that noise which is introduced by the sensors used to measure the output variables being fed back to the controller. The second, measurement matrix noise, is identical to output measurement noise, except that it is the noise associated with measuring the quantities required to augment a rank defective CB matrix. The third type of noise, disturbance noise, allows the user to add disturbance inputs directly into the state equations in the form

$$\dot{\underline{x}} = \underline{A}\underline{x} + \underline{B}\underline{u} + \underline{G}\underline{w} \quad (\text{A-1})$$

where \underline{w} is a vector of random variables representing the disturbance input and \underline{G} is a matrix that governs the distribution of the noise into the state equations.

This addition also provides the user the option of making multiple simulations to statistically determine the influence of noise through the use of a Monte Carlo analysis.

2. User's Guide. Option 25 is selected to enter the data for the simulation of noise. Prior to entry into option 25 the user must have provided the number of states, outputs and inputs, by option 2, 9, or 199. In this case there are 3 states, 2 inputs, and 2 outputs. The following

prompt appears upon selection of option 25:

```
OPTION, PLEASE > #  
? 25
```

THIS OPTION ALLOWS SIMULATION OF INDEPENDENT GAUSSIAN
DISTURBANCES AND SENSOR NOISE.

ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

```
ENTER,SUPPRESS OR RESET DISTURBANCE INPUT....."0"  
ENTER,SUPPRESS OR RESET OUTPUT MEASUREMENT NOISE..."1"  
ENTER,SUPPRESS OR RESET MEASUREMENT MATRIX NOISE..."2"  
DEFINE MONTE CARLO SIMULATION....."3"  
TO QUIT OPTION 25....."4"  
? 0
```

At the prompt the user selects "0" to operate on the disturbance noise. In this case the user desires to enter new noise data, makes the proper selection, and enters the data at the prompts.

THIS OPTION ALLOWS SIMULATION OF A DISTURBANCE OF THE
FORM $\dot{D} = AX + BU + GW$, WHERE W IS A VECTOR OF N
INDEPENDENT GAUSSIAN RANDOM VARIABLES.
 G IS A MATRIX THAT IS N BY N WHERE N IS THE NUMBER OF
STATES, FORMING A LINEAR COMBINATION OF THE RANDOM
VARIABLES.

ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

```
ENTER NEW DISTURBANCE PARAMETERS....."0"  
SUPPRESS DISTURBANCE INPUT....."1"  
RESET DISTURBANCE INPUT....."2"  
? 0  
ENTER THE MEAN AND STANDARD DEVIATION OF W(1)  
? 2.3  
W(2)  
? 2.3  
W(3)  
? 2.3  
ENTER THE G MATRIX BY ROW, 3 ELEMENTS PER ROW
```

```

ROW 1
? 1,2,3
ROW 2
? 2,3,1
ROW 3
? 3,1,2

```

Upon completion of the disturbance noise input, the program returns to the main menu for option 25 and awaits further input. At this point the user proceeds to input measurement matrix noise, output measurement noise, and define the size of the Monte Carlo analysis.

ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

```

ENTER,SUPPRESS OR RESET DISTURBANCE INPUT....."0"
ENTER,SUPPRESS OR RESET OUTPUT MEASUREMENT NOISE..."1"
ENTER,SUPPRESS OR RESET MEASUREMENT MATRIX NOISE..."2"
DEFINE MONTE CARLO SIMULATION....."3"
TO QUIT OPTION 25....."4"

```

```

? 1
THIS OPTION ALLOWS SIMULATION OF NOISY OUTPUT SENSORS,
CORRUPTING THE SIGNAL BEING FED BACK. INDEPENDENT
GAUSSIAN NOISE IS ADDED TO EACH ELEMENT OF THE OUTPUT
VECTOR WITH MEAN AND STANDARD DEVIATION OF YOUR CHOICE

```

ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

```

TO ENTER NEW OUTPUT NOISE PARAMETERS....."0"
TO SUPPRESS OUTPUT SENSOR NOISE....."1"
TO RESET OUTPUT SENSOR NOISE....."2"

```

```

? 0
ENTER THE MEAN AND STANDARD DEVIATION OF THE NOISE
ASSOCIATED WITH MEASURING OUTPUT 1:

```

```

? 2,3
OUTPUT2

```

```

? 1,2
ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

```

ENTER,SUPPRESS OR RESET DISTURBANCE INPUT....."0"
 ENTER,SUPPRESS OR RESET OUTPUT MEASUREMENT NOISE..."1"
 ENTER,SUPPRESS OR RESET MEASUREMENT MATRIX NOISE..."2"
 DEFINE MONTE CARLO SIMULATION....."3"
 TO QUIT OPTION 25....."4"

? 2

THIS OPTION ALLOWS SIMULATION OF A NOISY MEASUREMENT OF
 THE STATE DERIVATIVES IN THE CASE OF AN IRREGULAR PLANT
 THE NOISE IS MODELLED AS INDEPENDENT GAUSSIAN RANDOM
 VARIABLES WITH MEAN AND VARIANCE OF YOUR CHOICE ADDED TO
 ANY OR ALL OF THE DERIVATIVES OF THE X2 VECTOR

ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

TO ENTER NEW MEASUREMENT NOISE PARAMETERS....."0"
 TO SUPPRESS MEASUREMENT MATRIX NOISE....."1"
 TO RESET MEASUREMENT MATRIX NOISE....."2"

? 0

ENTER THE MEAN AND STANDARD DEVIATION OF THE NOISE
 ASSOCIATED WITH MEASURING STATE DERIVATIVE 1

? 5,4

STATE DERIVATIVE 2?

? 6,5

ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

ENTER,SUPPRESS OR RESET DISTURBANCE INPUT....."0"
 ENTER,SUPPRESS OR RESET OUTPUT MEASUREMENT NOISE..."1"
 ENTER,SUPPRESS OR RESET MEASUREMENT MATRIX NOISE..."2"
 DEFINE MONTE CARLO SIMULATION....."3"
 TO QUIT OPTION 25....."4"

? 3

ENTER NUMBER OF SIMULATION RUNS DESIRED FOR MONTE CARLO
 ANALYSIS.....>

? 5

The user has selected a Monte Carlo simulation that is to
 be comprised of five independent noise simulations. Now,
 every time the simulation option (option 26) is selected
 the user will be asked if that simulation is to be included
 in the Monte Carlo analysis. The user may exit MULTI,
 log off, or run as many "non-Monte Carlo" simulations as

he chooses. Once five Monte Carlo simulations have been executed the program returns to normal. To avoid unnecessary expenditure of computer resources, the total number of Monte Carlo simulations is limited to twenty-five. Finally, the user decides that the output measurement noise should be suppressed temporarily. This allows the noise to be eliminated without having to re-enter the noise parameters when the noise input is required. Entering a "1" at the prompt suppresses the noise, entering a "2" will reset suppressed noise. After suppressing the noise the user exits to the main program.

ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

ENTER,SUPPRESS OR RESET DISTURBANCE INPUT....."0"
 ENTER,SUPPRESS OR RESET OUTPUT MEASUREMENT NOISE..."1"
 ENTER,SUPPRESS OR RESET MEASUREMENT MATRIX NOISE..."2"
 DEFINE MONTE CARLO SIMULATION....."3"
 TO QUIT OPTION 25....."4"

? :
 THIS OPTION ALLOWS SIMULATION OF NOISY OUTPUT SENSORS, CORRUPTING THE SIGNAL BEING FED BACK. INDEPENDENT GAUSSIAN NOISE IS ADDED TO EACH ELEMENT OF THE OUTPUT VECTOR WITH MEAN AND STANDARD DEVIATION OF YOUR CHOICE

ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

TO ENTER NEW OUTPUT NOISE PARAMETERS....."0"
 TO SUPPRESS OUTPUT SENSOR NOISE....."1"
 TO RESET OUTPUT SENSOR NOISE....."2"

? :
 ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:

```

ENTER,SUPPRESS OR RESET DISTURBANCE INPUT....."0"
ENTER,SUPPRESS OR RESET OUTPUT MEASUREMENT NOISE..."1"
ENTER,SUPPRESS OR RESET MEASUREMENT MATRIX NOISE..."2"
DEFINE MONTE CARLO SIMULATION....."3"
TO QUIT OPTION 25....."4"
? 4

```

At this point the user desires to verify the inputs he made in option 25. This is accomplished with option 125, which displays the current noise parameters. Notice under the data for output measurement noise the word "(SUPPRESSED)", indicating that this noise is not currently being used.

```

OPTION, PLEASE > #
? 125

```

DISTURBANCE NOISE PARAMETERS
3 MATRIX

.1000E+01	.2000E+01	.3000E+01
.2000E+01	.3000E+01	.1000E+01
.3000E+01	.1000E+01	.2000E+01

NOISE MEANS AND STANDARD DEVIATIONS

.2000E+01	.3000E+01
.2000E+01	.3000E+01
.2000E+01	.3000E+01

OUTPUT MEASUREMENT NOISE
MEAN AND STANDARD DEVIATION

.2000E+01	.3000E+01
.1000E+01	.2000E+01
(SUPPRESSED)	

```

MEASUREMENT MATRIX NOISE
MEAN AND STANDARD DEVIATION
.5000E+01 .4000E+01
.6000E+01 .5000E+01

```

3. Programmer's Guide. The following FORTRAN code is located in PROGRAM OPT20, a subprogram of the executive program MULTI. This portion of the noise option addition is an interactive routine in which the user enters the desired noise, associated parameters, and the number of runs desired in the Monte Carlo analysis. The following variables have been introduced in this section of code:

<u>Variable</u>	<u>Description</u>
WRMEAN(I)	Vector containing the means of each disturbance to be added to the state equations.
WSIGMA(I)	Vector containing standard deviations of disturbances.
G(I,J)	Matrix distributing disturbances into state equation.
DISTURB	Integer flag indicating existence of disturbance noise.
PG(I,J)	Matrix where G(I,J) is permanently stored when the disturbance noise is suppressed.

RMEAN(I)	Vector of output measurement noise means.
RSIGMA(I)	Vector of output measurement noise standard deviations.
PRMN(I)	Vector where RMEAN(I) is permanently stored when the output measurement noise is suppressed.
PSIG(I)	Vector where RSIGMA(I) is permanently stored.
NOISE	Integer flag indicating existence of output measurement noise.
MRMEAN(I)	Vector of measurement matrix noise means.
MSIGMA(I)	Vector of measurement matrix noise standard deviations.
PMRMN(I)	Vector where MRMEAN(I) permanently stored when measurement matrix noise is suppressed.
PMSIG(I)	Vector where MSIGMA(I) is permanently stored.
MNOISE	Integer flag indicating existence of measurement matrix noise.
MONTC	Integer indicating number of Monte Carlo simulations desired.
DAT4	=80, output device assignment for local file MEM30, the file that contains the running sum of the simulation data.
MCOUNT	Integer counter indicating the number of Monte Carlo simulations already run. This variable is only initialized and stored in this part of the code.

```

*****
C***** OPTION 25 IS THE NOISE INPUT OPTION*****
C*****
2025 PRINT*, 'THIS OPTION ALLOWS SIMULATION OF INDEPENDENT GAUSSIAN'
      PRINT*, 'DISTURBANCES AND SENSOR NOISE.'
      PRINT*, ''
      IF (IFLAG(2).EQ.0) THEN
        PRINT*, '# OF STATES, INPUTS & OUTPUTS MISSING...SEE OPTION #2'
        GO TO 8007
      ENDIF
5000 PRINT*, 'ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:'
      PRINT*, ''
      PRINT*, ''
      PRINT*, 'ENTER,SUPPRESS OR RESET DISTURBANCE INPUT....."0"'
      PRINT*, 'ENTER,SUPPRESS OR RESET OUTPUT MEASUREMENT NOISE..."1"'
      PRINT*, 'ENTER,SUPPRESS OR RESET MEASUREMENT MATRIX NOISE..."2"'
      PRINT*, 'DEFINE MONTE CARLO SIMULATION....."3"'
      PRINT*, 'TO QUIT OPTION 25....."4"'
      READ*,ISKIP
      IF (ISKIP.EQ.4) GO TO 8007
      IFLAG(25)=1
      IF (ISKIP.EQ.0) THEN
        PRINT*, 'THIS OPTION ALLOWS SIMULATION OF A DISTURBANCE OF THE'
        PRINT*, 'FORM  $\dot{X} = AX + BU + GW$ , WHERE W IS A VECTOR OF N'
        PRINT*, 'INDEPENDENT GAUSSIAN RANDOM VARIABLES.'
        PRINT*, 'G IS A MATRIX THAT IS N BY N WHERE N IS THE NUMBER OF'
        PRINT*, 'STATES, FORMING A LINEAR COMBINATION OF THE RANDOM'
        PRINT*, 'VARIABLES.'
        PRINT*, ''
        PRINT*, 'ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:'
        PRINT*, ''
        PRINT*, ''
        PRINT*, 'ENTER NEW DISTURBANCE PARAMETERS....."0"'
        PRINT*, 'SUPPRESS DISTURBANCE INPUT....."1"'
        PRINT*, 'RESET DISTURBANCE INPUT....."2"'
        READ*, ISKIP
        IF (ISKIP.EQ.0) THEN
          C-----ENTER DISTURBANCE PARAMETERS-----
          PRINT*, 'ENTER THE MEAN AND STANDARD DEVIATION OF W:'
          READ*, WRMEAN(1),WSIGMA(1)
          IF (N.EQ.1) GO TO 5002
          DO 5001 I=2,N
            PRINT*, 'W(',I,')'
5001    READ*, WRMEAN(I),WSIGMA(I)
5002    PRINT*, 'ENTER THE G MATRIX BY ROW, ',N,' ELEMENTS PER ROW'
          DO 5004 I=1,N
            PRINT*, 'ROW ',I
            READ*, (G(I,J),J=1,N)
            DO 5003 J=1,N

```

```

5003 PG(I,J)=G(I,J)
5004 CONTINUE
      DISTURB=1
      GO TO 5000
    ENDIF
C *****
C*****SUPPRESS DISTURBANCE PARAMETERS*****
      IF (ISKIP.EQ.1) THEN
        DO 5007 I=1,N
        DO 5006 J=1,N
5006   G(I,J)=0
5007   CONTINUE
        DISTURB=0
        GO TO 5000
      ENDIF
C*****RESET DISTURBANCE PARAMETERS*****
      IF (ISKIP.EQ.2) THEN
        DO 5010 I=1,N
        DO 5009 J=1,N
5009   G(I,J)=PG(I,J)
5010   CONTINUE
        DISTURB=1
      ENDIF
      GO TO 5000
    ENDIF
C*****
C****          OUTPUT MEASUREMENT NOISE          ****
C*****
      IF (ISKIP.EQ.3) THEN
        PRINT*, 'THIS OPTION ALLOWS SIMULATION OF NOISY OUTPUT SENSORS.'
        PRINT*, 'CORRUPTING THE SIGNAL BEING FED BACK. INDEPENDENT'
        PRINT*, 'GAUSSIAN NOISE IS ADDED TO EACH ELEMENT OF THE OUTPUT'
        PRINT*, 'VECTOR WITH MEAN AND STANDARD DEVIATION OF YOUR CHOICE'
        PRINT*, '
        PRINT*, 'ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:
        PRINT*, '
        PRINT*, '
        PRINT*, 'TO ENTER NEW OUTPUT NOISE PARAMETERS.....0'
        PRINT*, 'TO SUPPRESS OUTPUT SENSOR NOISE.....1'
        PRINT*, 'TO RESET OUTPUT SENSOR NOISE.....2'
        READ*, ISKIP
C*****ENTER OUTPUT NOISE*****
      IF (ISKIP.EQ.0) THEN
        PRINT*, 'ENTER THE MEAN AND STANDARD DEVIATION OF THE NOISE'
        PRINT*, 'ASSOCIATED WITH MEASURING OUTPUT 1:'
        READ*, RMEAN(1),RSIGMA(1)
        PRMN(1)=RMEAN(1)
        PSIG(1)=RSIGMA(1)

```

```

DO 5012 I=2,P
PRINT *, 'OUTPUT',I
READ*, RMEAN(I),RSIGMA(I)
PRMN(I)=RMEAN(I)
PSIG(I)=RSIGMA(I)
5012 CONTINUE
NOISE=1
GO TO 5000
ENDIF
C***** SUPPRESS OUTPUT NOISE *****
IF (ISKIP.EQ.1) THEN
DO 5014 I=1,P
RMEAN(I)=0
5014 RSIGMA(I)=0
NOISE=0
GO TO 5000
ENDIF
C***** RESET OUTPUT NOISE *****
IF (ISKIP.EQ.2) THEN
DO 5016 I=1,P
RMEAN(I)=PRMN(I)
5016 RSIGMA(I)=PSIG(I)
NOISE=1
ENDIF
GO TO 5000
ENDIF
C*****
C**** MEASUREMENT MATRIX NOISE ****
C*****
IF (ISKIP.EQ.2) THEN
PRINT*, 'THIS OPTION ALLOWS SIMULATION OF A NOISY MEASUREMENT OF
PRINT*, 'THE STATE DERIVATIVES IN THE CASE OF AN IRREGULAR PLANT
PRINT*, 'THE NOISE IS MODELLED AS INDEPENDENT GAUSSIAN RANDOM
PRINT*, 'VARIABLES WITH MEAN AND VARIANCE OF YOUR CHOICE ADDED TO
PRINT*, 'ANY OR ALL OF THE DERIVATIVES OF THE X VECTOR
PRINT*,
PRINT*, 'ENTER YOUR CHOICE OF THE FOLLOWING OPTIONS:
PRINT*,
PRINT*,
PRINT*, 'TO ENTER NEW MEASUREMENT NOISE PARAMETERS....."0"'
PRINT*, 'TO SUPPRESS MEASUREMENT MATRIX NOISE....."1"'
PRINT*, 'TO RESET MEASUREMENT MATRIX NOISE....."2"'
READ*, ISKIP
C***** ENTER MEASUREMENT MATRIX NOISE*****
IF (ISKIP.EQ.0) THEN
PRINT*, 'ENTER THE MEAN AND STANDARD DEVIATION OF THE NOISE'
PRINT*, 'ASSOCIATED WITH MEASURING STATE DERIVATIVE 1'
READ *, MRMEAN(1), MSIGMA(1)

```

```

        PMRMN(1)=MRMEAN(1)
        PMSIG(1)=MSIGMA(1)
        DO 5018 I=2,P
        PRINT*, 'STATE DERIVATIVE ',I,'?'
        READ*, MRMEAN(I),MSIGMA(I)
        PMRMN(I)=MRMEAN(I)
5018    PMSIG(I)=MSIGMA(I)
        MNOISE=1
        GO TO 5000
    ENDIF
C***** SUPPRESS MEASUREMENT NOISE *****
    IF (ISKIP.EQ.1) THEN
        DO 5020 I=1,P
        MRMEAN(I)=0
        MSIGMA(I)=0
5020    CONTINUE
        MNOISE=0
        GO TO 5000
    ENDIF
C***** RESET MEASUREMENT NOISE *****
    IF (ISKIP.EQ.2) THEN
        DO 5022 I=1,P
        MRMEAN(I)=PMRMN(I)
        MSIGMA(I)=PMSIG(I)
5022    CONTINUE
        MNOISE=1
        ENDIF
        GO TO 5000
    ENDIF
C ***** DEFINE MONTE CARLO ANALYSIS *****
    IF (ISKIP.EQ.3) THEN
5023    PRINT*, 'ENTER NUMBER OF SIMULATION RUNS DESIRED FOR MONTE CARLO'
        PRINT*, 'ANALYSIS.....>'
        READ*, MONTC
        IF (MONTC.GT.25) THEN
            PRINT*, 'YOU HAVE GOT TO BE JOKING. OBVIOUSLY YOUR NOT PAYING'
            PRINT*, 'FOR THIS. THE OUTPUT WILL BE ROUTED TO THE IG FOR'
            PRINT*, 'FRAUD, WASTE & ABUSE INVESTIGATION.'
            PRINT*,
            GO TO 5023
        ENDIF
        DAT4=80
        OPEN (DAT4,FILE='MEM30')
        REWIND DAT4
        MCOUNT = 0
        WRITE (DAT4,*) MCOUNT
        CLOSE (DAT4,STATUS='KEEP')
        GO TO 5000
    ENDIF

```

The noise described by the entries in option 25 is entered into the simulation in option 26 by means of several subroutine calls to SUBROUTINE GPNML (listed below). This subroutine generates a seed and then makes a call to the IMSL Library routine GGNMC which returns the random vector RDEV(I). Variables introduced here are:

<u>Variable</u>	<u>Description</u>
DSEED	Real seed for IMSL routine GGNML.
RDEV(I)	Vector, random and zero-mean, returned by GGNML.
NR	Integer dimension of RDEV(I).

```

      SUBROUTINE GPNML (RMEAN,RSIGMA,N,RDEV)
C   DEVIATES RETURNED FROM IMSL IN RDEV(I)
      INTEGER NR
      REAL MRMEAN,MSIGMA
      COMMON /B 27/ MONTC,MCOUNT
      DIMENSION RMEAN(15),RSIGMA(15),RDEV(15)
      DOUBLE PRECISION DSEED
      DATA DSEED /2001.00/
      NR=N
C   REPETITIVE CALLS TO GGNMC (IMSL) WILL AUTOMATICALLY CHANGE DSEED
C   GGNML RETURNS A NORMALIZED ZERO MEAN GAUSSIAN N(0,1)
      DSEED = DSEED + (1000 + MCOUNT)
      CALL GGNML (DSEED,NR,RDEV)
      DO 5025 I=1,N
5025 RDEV(I)=RDEV(I)*RSIGMA(I) + RMEAN(I)
C   TRANSFORM THE NORMALIZED VECTOR TO N(RMEAN,RSIGMA)
      RETURN
      END

```

A number of minor changes were made throughout MULTI to accommodate the noise input option. Previously, the calculation step size was entered in option 25. This function is now accomplished in option 24. Option 125 now

prints out the current values of the noise parameters. All noise data entered in option 25 is stored in local file MEM20, and as a result the options which affect the reading and writing of MEM20 (options 29, 99, and 199) are changed accordingly. Finally, option 26 includes a section of code that reads, operates on, and writes to local file MEM30 to keep a running total of the simulation data needed to perform the Monte Carlo analysis. During each simulation the current data is added to the values stored in MEM30 from previous simulations, creating a running total at each time increment. When the last run is complete, the running totals are divided by the total number of runs to obtain an "average" run. This data can then be plotted in the same manner as the results of any other simulation. Currently this code, listed below, only calculates the mean value of multiple simulation runs. It is recommended that in the future this be expanded to include a calculation of the standard deviation as well. The new variables in this section are:

<u>Variable</u>	<u>Description</u>
MONTY	Logical character indicating whether user wishes current simulation to be included in the Monte Carlo analysis.
DATD	=90, input device assignment for local file MEM30.

DATA4 =80, output device assignment for local file
MEM30.

MYP(IJ,I) Matrix containing a running sum of output
data.

MUP(IJ,I) Matrix containing a running sum of input (U)
data.

MVP(IJ,I) Matrix containing a running sum of input (V)
data.

MXP(IJ,I) Matrix containing a running sum of state data.

B. Custom Input Option

1. Description. This option expands the input alternatives to include a wide variety of possibilities as defined by the user. By selecting the custom input feature of option 22, the user can select ten points that define the input magnitude as a function of time. The points are connected with straight lines by the program and if desired the corners are smoothed. The option of using the original input routine is retained and its use is recommended whenever possible, since it is easier to use.

2. User's Guide. To select a custom input, the user enters "22" at the option prompt. Following is a sample of the interactive prompts and inputs.

OPTION, PLEASE > #
? 22

THIS OPTION SETS THE INPUT COMMAND VECTOR, V

DO YOU WANT THE STANDARD OR CUSTOM INPUT?
ENTER S OR C >

? C

THIS PORTION OF THE PROGRAM ALLOWS THE USER
TO DEFINE 10 POINTS ALONG A CUSTOM INPUT THAT
ARE TO BE CONNECTED BY STRAIGHT LINES AND
THEN SMOOTHED IF SO DESIRED.

ENTER INPUT 1: TIME, MAGNITUDE>

PT. 1>>

? 1,1

PT. 2>>

? 2,2

PT. 3>>

? 3,3

PT. 4>>

? 4,4

PT. 5>>

? 5,5

PT. 6>>

? 6,6

PT. 7>>

? 7,7

PT. 8>>

? 8,8

PT. 9>>

? 9,9

PT. 10>>

? 10,10

ENTER INPUT 2: TIME, MAGNITUDE>

PT. 1>>

? 2,3

PT. 2>>

? 3,4

PT. 3>>

? 5,6

PT. 4>>

? 4,3

TIME FOR PT. 4 MUST BE GREATER THAN
OR EQUAL TO PT. 3. TRY AGAIN.

ENTER INPUT 2: TIME, MAGNITUDE>

PT. 1>>

? 2,3

PT. 2>>

? 3,4

PT. 3>>

? 5,6

PT. 4>>

? 7,8

```

PT. 5>>
? 9,10
PT. 6>>
? 11,12
PT. 7>>
? 13,14
PT. 8>>
? 15,16,
PT. 9>>
? 17,18
PT. 10>>
? 19,20
DO YOU WANT TO SMOOTH THE INPUTS? Y OR N
? Y

```

Notice that if the user attempts to enter the data in other than sequential or chronological order, the program interprets this as going backward in time and requests corrected data. Like most of the data options in MULTI, the values may be verified in its corresponding 100-series option.

```

OPTION, PLEASE / #
1 100

(10). INITIAL STATES...

.0000E+00 .0000E+00 .0000E+00

2(0). INITIAL STATES...

.0000E+00 .0000E+00
CUSTOM INPUT
INPUT 1
PT. TIME MAG
1 1. 1.
2 2. 2.

```

3	3.	3.
4	4.	4.
5	5.	5.
6	6.	6.
7	7.	7.
8	8.	8.
9	9.	9.
10	10.	10.
INPUT 2		
PT.	TIME	MAG
1	2.	3.
2	3.	4.
3	5.	6.
4	7.	8.
5	9.	10.
6	11.	12.
7	13.	14.
8	15.	16.
9	17.	18.
10	19.	20.
INPUT IS SMOOTHED		

OPTION. PLEASE > #

In order to make effective use of the custom input feature it is imperative that the user understand the mathematical foundations of the smoothing routine and the assumptions made in implementing the option. The specifics of the smoothing algorithm are discussed in the programmer's guide. Following is a summary of features and limitations that the user may find useful.

a. Step inputs cannot be smoothed. It is assumed that if a smoothed input is desired a ramp would be selected for the initial step up or down. The program defines a step input as any two consecutive points having the same time axis coordinate. If any part of any of the

inputs is a step, then none of the inputs can be smoothed. If the user desires smoothed step inputs within the custom input, it is recommended that a ramp with a duration of less than a sample period be entered. It is very likely that a ramp of such short duration cannot be smoothed with the polynomial techniques used, but even if unable to smooth the step the algorithm will continue to smooth the remainder of the input normally.

b. It is important that the input be defined for at least the longest simulation time anticipated. In most cases, failure to do so will result in the value of the magnitude of the last point being held throughout the undefined region. Obviously, points beyond the simulation time will never be encountered in option 26 but they may be useful for shaping the input prior to the end of the simulation.

c. All ten points must be defined. Note that there are no trivial inputs. If no input to a particular channel is desired, then an input that is specified as zero magnitude for the entire simulation time is required, that is, at each of the ten points. (Simply entering zeros at both the time and magnitude prompts will result in an input which is only defined at the origin.)

d. The input always begins at the origin. Unless a step is desired, the time at point 1 should be

greater than zero. If, however, the time at point 1 is chosen to be zero, the magnitude should be non-zero.

e. Clever application of the mathematical principles used for the smoothing algorithm can produce nearly any input desired. The duration and amount of smoothing can be varied without changing the basic input by inserting extra points along straight line segments. A sample input, both smoothed and unsmoothed, is shown in Figures A.1 and A.2.

3. Programmer's Guide. The bulk of the code to accomplish this feature is located in one of two places--in PROGRAM OPT20 under option 22, and in PROGRAM OPT26. The code in option 22 is where the data is entered for the custom input feature, and where the parameters for the smoothing curve are calculated. The basic structure of the algorithm, as shown in Figure A.3, is to first establish the unsmoothed, "dot-to-dot" input curve. Then, if smoothing is desired, a third order polynomial is chosen such that the slope and magnitude of the polynomial match the basic curve at the beginning and end of smoothing. Smoothing occurs in the last 20 percent of the line segment before the point of interest and the first 20 percent of the line segment following the point. Often, the curve to be smoothed changes slope too rapidly to be adequately smoothed by a third order polynomial.

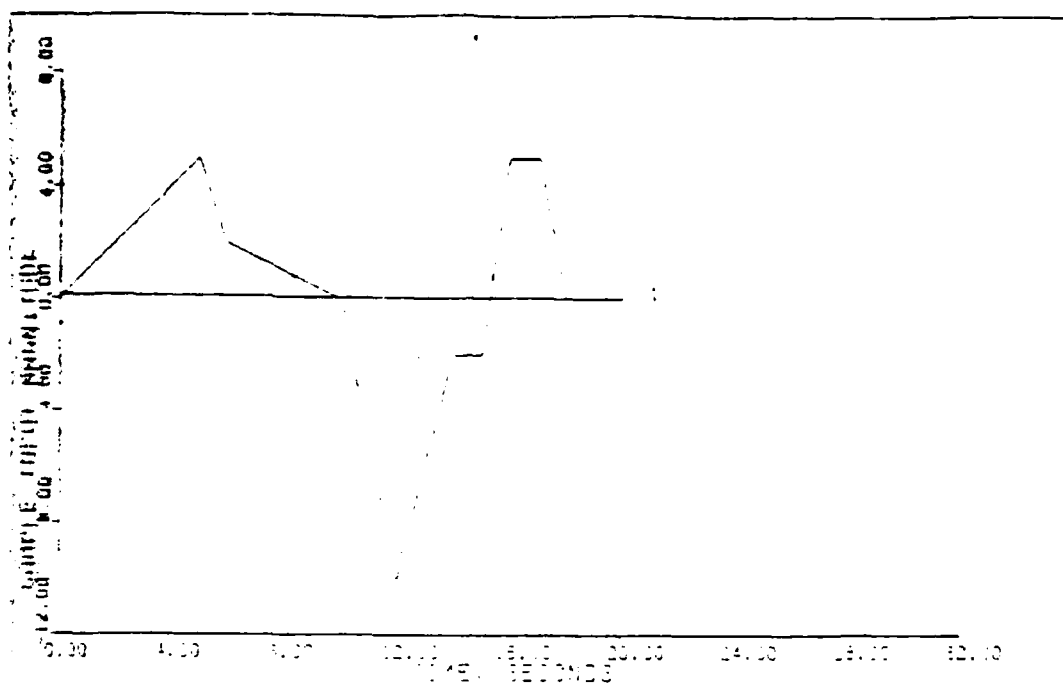


Fig. A.1. Sample Custom Input--Unsmoothed

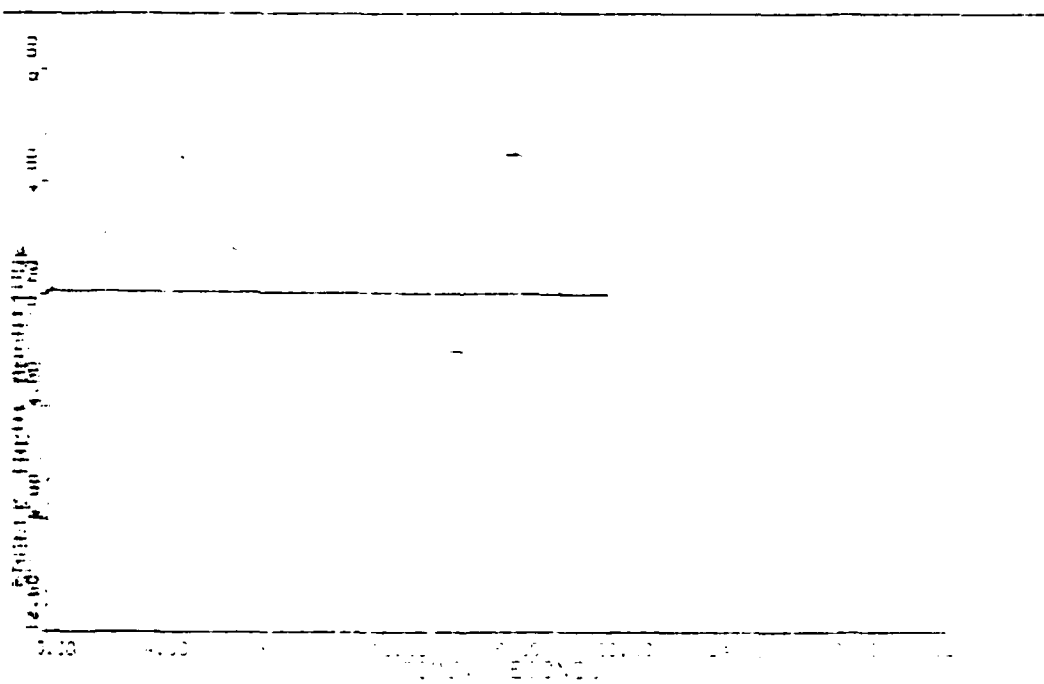


Fig. A.1. Sample Custom Input--Smoothed

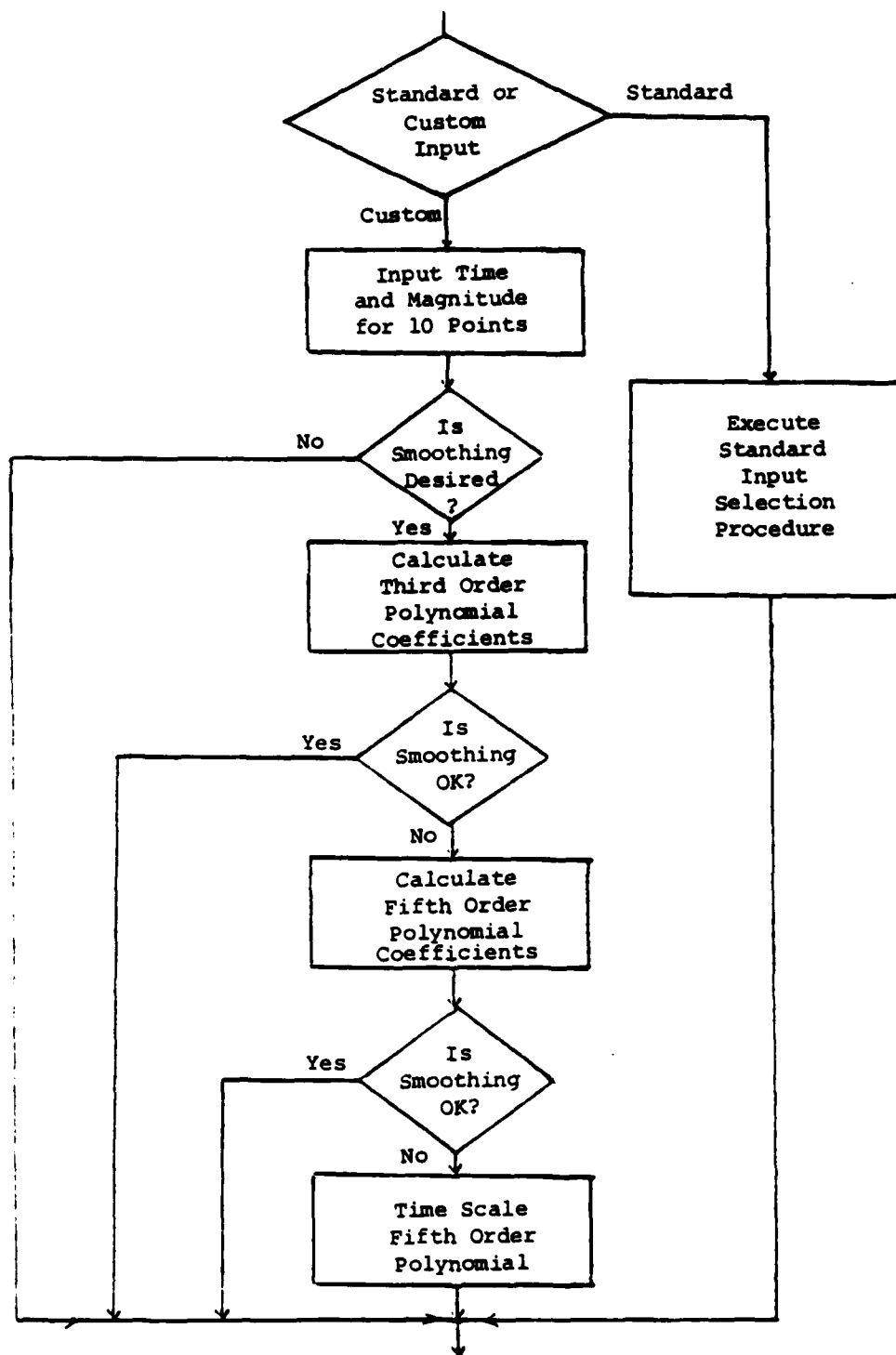


Fig. A.3. Option #22 Algorithm Outline

This condition is indicated by the second derivative of the polynomial at the beginning and end of smoothing being opposite in sign from the desired slope change. This situation can sometimes be corrected by using a fifth order polynomial and improving two more conditions; specifically that the acceleration at the beginning and end of smoothing be continuous (i.e., zero). Even fifth order polynomials can have unacceptable smoothing characteristics if the third derivative at the beginning or end of smoothing is of opposite sign of the desired acceleration change. In this event the algorithm attempts to time scale the fifth order polynomial to satisfy these conditions. If the user attempts to smooth an input that has large slope changes with short line segments the smoothing will be unsatisfactory. The only way to identify inadequate smoothing is to plot the inputs (V vector) using a calcomp plot option (34 or 35). The astute programmer will find that the routine used to find an acceptable time scaling factor is an unsophisticated, brute force sequential search. It was found that determining that the current value does not satisfy the required conditions yields no information on which direction to search, rendering a more efficient approach, like a binary search, impossible. If no solution is found, eventually a matrix that must be inverted becomes singular, and the program returns a message to that effect and does not smooth that particular point. The math used

to solve these problems is not complex, but is difficult to follow from the code alone. The key equations are developed below to facilitate understanding of the program. The variables used in this development are as follows:

t independent variable, time
 h dependent variable, input magnitude
 t_0 time at previous input point
 t_1 time at current point
 t_2 time at next point
 \hat{t} time smoothing begins
 $\hat{\Delta t}$ duration of smoothing
 a_n ($n=0,5$) coefficients of smoothing polynomial
 h_0 magnitude of previous input point
 h_1 magnitude of current point
 h_2 magnitude of next point
 s_1 slope before current point
 s_2 slope after current point
 f time scaling factor

a. Third order smoothing. If a third order polynomial is to be used, the general form of the input between \hat{t} and $(\hat{t} + \hat{\Delta t})$ is:

$$h(t) = a_0 + a_1(t - \hat{t}) + a_2(t - \hat{t})^2 + a_3(t - \hat{t})^3 \quad (A-2)$$

To solve for the four unknowns (a_0 , a_1 , a_2 , and a_3), four constraints must be satisfied. In this case, the conditions

are chosen in order to match the magnitude and slope of the straight-line input at the beginning and end of smoothing ($t = \hat{t}$, and $t = \hat{t} + \hat{\hat{t}}$). Applying these conditions to the polynomial and its derivative (slope) and then solving the four simultaneous equations yields:

$$a_0 = h_0 + s_1(\hat{t} - t_0) \quad (A-3)$$

$$a_1 = s_1 \quad (A-4)$$

$$a_2 = [3(h_1 + .2t_2s_2 - .2t_1s_2 - a_0 - a_1\hat{t}) - s_2\hat{t} + a_1\hat{t}] / \hat{t}^2 \quad (A-5)$$

$$a_3 = (s_2 - a_1 - 2a_2\hat{t}) / 3\hat{t}^2 \quad (A-6)$$

b. Fifth order smoothing. The general form of the fifth order polynomial used to smooth more difficult inputs is:

$$h(t) = a_0 + a_1(t - \hat{t}) + a_2(t - \hat{t})^2 + a_3(t - \hat{t})^3 + a_4(t - \hat{t})^4 + a_5(t - \hat{t})^5 \quad (A-7)$$

Since there are now six unknowns, two more constraints must be applied to find a unique solution for each of the coefficients. These conditions are chosen so that the second derivative of the input is zero at the beginning and end of smoothing. These constraints result in the following equations:

$$a_0 = h_0 + s_1(\hat{t}-t_0) \quad (A-8)$$

$$a_1 = s_1 \quad (A-9)$$

$$a_2 = 0 \quad (A-10)$$

$$a_3 + a_4\hat{t} + a_5\hat{t}^2 = \{h_1 + s_2[.2(t_2-t_1)] - a_0 - a_1\hat{t}\}/\hat{t}^3 \quad (A-11)$$

$$a_3 + (4/3)a_4\hat{t} + (5/3)a_5\hat{t}^2 = (s_2 - a_1)/3\hat{t}^2 \quad (A-12)$$

$$a_3 + 2a_4\hat{t} + (20/6)a_5\hat{t}^2 = 0 \quad (A-13)$$

For programming convenience, since a_2 is always zero, a_3 , a_4 , and a_5 are changed to a_2 , a_3 , and a_4 respectively. Equations (A-11), (A-12), and (A-13) are solved as simultaneous equations by MULTI in PROGRAM OPT20 under option #22.

c. Time scaled fifth order polynomial. The general form of the time scaled fifth order polynomial is the same except for the independent variable:

$$h(t) = a_0 + a_1\bar{t} + a_2\bar{t}^3 + a_3\bar{t}^4 + a_4\bar{t}^5 \quad (A-14)$$

$$\bar{t} = (t - \hat{t})/f \quad (A-15)$$

The conditions of the fifth order polynomial are again applied with the additional constraint that the third derivative at the beginning of the smoothing be the same

sign as the change in slope desired and of opposite sign at the end of smoothing. The program calculates the polynomial coefficients and executes an iterative search for a value of "f" that will satisfy these constraints.

d. FORTRAN code. The equations presented in the last three paragraphs are only the basic framework for the custom input routine. The programmer will also notice a number of conditional statements in both option 22 and option 26 to avoid overflow conditions that result from dividing by zero and other discontinuities. The variables introduced in these sections of code are contained in the comment statement preceding option 22.

```

ELSEIF OPTINP.EQ. 22 THEN
  PRINT*, 'THIS PORTION OF THE PROGRAM ALLOWS THE USER'
  PRINT*, 'TO DEFINE 10 POINTS ALONG A CUSTOM INPUT THAT'
  PRINT*, 'ARE TO BE CONNECTED BY STRAIGHT LINES AND'
  PRINT*, 'THEN SMOOTHED IF SO DESIRED.'
  DO 1950 K=1,P
1951  PRINT*, 'ENTER INPUT (K,1) TIME, MAGNITUDE'
    DO 1952 I=1,10
      PRINT*, 'PT. (I,1)'
      F=1
      READ*, (INPPT(K,I,J),J=1,2)
      IF (INPPT(K,I,1).LT.0. OR INPPT(K,I,1).AND.F.GE.1) THEN
        PRINT*, 'TIME FOR PT. (I,1) MUST BE GREATER THAN'
        PRINT*, 'OR EQUAL TO PT. (I,1). TRY AGAIN.'
        GO TO 1951
      ENDIF
1952  CONTINUE
1953  CONTINUE
    PRINT*, 'DO YOU WANT TO SMOOTH THE INPUTS? Y OR N'
    READ*(A), SMOPT
  C*****
  C    THE CODE THAT FOLLOWS COMPUTES THE COEFFICIENTS OF THE
  C    POLYNOMIALS THAT ARE USED AS SMOOTHING CURVES BETWEEN LINE SEGMENTS
  C    IN EACH INPUT. THE POLYNOMIALS ARE INITIALLY CHOSEN AS THIRD ORDER

```

```

C POLYNOMIALS SUCH THAT THEIR MAGNITUDE AND SLOPE MATCH THE LINE *
C SEGMENT VALUES AT THE POINTS WHERE THE SMOOTHING STARTS AND STOPS. *
C OFTEN, HOWEVER, THE INPUT CHANGES SLOPE TOO RAPIDLY TO BE SMOOTHED *
C BY A THIRD ORDER POLYNOMIAL. MATHEMATICALLY IT IS POSSIBLE TO MEET *
C ANY BOUNDARY CONDITIONS (OTHER THAN INFINITE SLOPES) WITH ONLY A *
C THIRD ORDER POLYNOMIAL BUT THE CURVE SOMETIMES INITIALLY TURNS IN *
C THE WRONG DIRECTION. THIS PHENOMENON IS EVIDENT IN THE SECOND *
C DERIVATIVE OF THE FUNCTION, WHICH SHOULD AT LEAST BE THE SAME SIGN *
C AS THE CHANGE IN SLOPE AT THE POINT OF INTEREST. THE CODE TESTS FOR *
C THIS CONDITION AND CALCULATES COEFFICIENTS FOR FIFTH ORDER POLYS *
C IF NECESSARY. THIS ALLOWS TWO MORE CONDITIONS TO BE IMPOSED ON THE *
C SMOOTHING CURVE, AND THEY ARE CHOSEN SUCH THAT THE ACCELERATION IS *
C CONTINUOUS AND ZERO AT THE START AND STOP OF THE SMOOTHING. IT IS *
C POSSIBLE IN EXTREMELY DIFFICULT SMOOTHING SITUATIONS (LIKE LARGE *
C SLOPE CHANGES IN VERY SHORT TIME PERIODS) THAT THE SAME PROBLEM WILL *
C ARISE IN THE THIRD DERIVATIVE (JERK) OF THE FIFTH ORDER POLYNOMIAL. *
C THIS CONDITION IS TESTED AS WELL AND IF NECESSARY THE FIFTH ORDER *
C POLYNOMIAL IS TIME SCALED BY A FACTOR 'F'. SINCE THE SCALE FACTOR *
C RESULTS IN NON-LINEAR SIMULTANEOUS EQUATIONS A SOLUTION (NOT UNIQUE) *
C IS FOUND THROUGH AN ITERATIVE SEARCH. A SOLUTION MAY NOT EXIST IN *
C THE REGION SEARCHED ( F > .01 ) *
C IN THIS EVENT THAT PARTICULAR POINT IS NOT SMOOTHED. NOTE THAT AS *
C THE ORDER OF THE POLYNOMIAL INCREASES IT APPROACHES A TAYLOR *
C SERIES REPRESENTATION OF THE INPUT, AND LESS SMOOTHING OCCURS. *
C AFTER SMOOTHING THE INPUT SEGMENTS ARE STRAIGHT IN THE MIDDLE 50% *
C OF THEIR LENGTH. AT EACH POINT THE INPUT IS SMOOTHED FOR 20% OF THE *
C LINE SEGMENT BEFORE AND 20% OF THE LINE SEGMENT FOLLOWING THE POINT *
C
C VARIABLES IN THIS SECTION OF CODE ARE: *
C
C K.....COUNTER, USUALLY THE CURRENT INPUT *
C I.....COUNTER, USUALLY THE CURRENT PT. IN INPUT K *
C J.....COUNTER, INTERNAL TO ARRAY INPPT *
C INPPT(K,I,J).....ARRAY, CONTAINING INPUT PTS, POLYNOMIAL *
C                   COEFFICIENTS, AND A SMOOTHING FLAG J=9 *
C T0.....INPPT(K,I-1,1), TIME AT LAST INPUT PT. *
C T1.....INPPT(K,I,1), TIME AT CURRENT INPUT PT. *
C T2.....INPPT(K,I+1,1), TIME AT NEXT INPUT PT. *
C H0.....INPPT(K,I-1,2), MAG. AT LAST INPUT PT. *
C H1.....INPPT(K,I,2), MAG. AT CURRENT INPUT PT. *
C H2.....INPPT(K,I+1,2), MAG. AT NEXT INPUT PT. *
C S1.....SLOPE UP TO CURRENT INPUT PT. *
C S2.....SLOPE AFTER CURRENT INPUT PT. *
C TH.....TIME SMOOTHING STARTS. *
C THH.....DURATION OF SMOOTHING *
C A0TH.....VALUE OF SMOOTHED ACCELERATION AT T=(TH+THH) *
C J0TH.....VALUE OF SMOOTHED JERK AT T=(TH+THH) *
C F.....TIME SCALING FACTOR F=1.0 *
C A0,A1,A2,A3,A4.....INPPT(K,I,3) J=3,7, POLYNOMIAL COEFFICIENTS *

```

```

C      INPPT(K,I,8).....= 0, WHEN UNABLE TO SMOOTH          *
C                                     = 3, WHEN THIRD ORDER SMOOTHING USED      *
C                                     = 5, WHEN FIFTH ORDER SMOOTHING USED      *
C                                     = F, WHEN FIFTH ORDER IS TIME SCALED      *
C*****
      IF (SMOPT.EQ.'Y') THEN
        DO 1855 K=1,P
          DO 1854 I=1,9
            IF (I.EQ.1) THEN
              H0=0
              T0=0
            ELSE
              H0=INPPT(K,I-1,2)
              T0=INPPT(K,I-1,1)
            ENDIF
            H1=INPPT(K,I,2)
            H2=INPPT(K,I+1,2)
            T1=INPPT(K,I,1)
            T2=INPPT(K,I+1,1)
            IF (T1.EQ.T2.OR.T0.EQ.T1) THEN
              PRINT*, 'YOU HAVE A STEP IN INPUT ',K,' THAT'
              PRINT*, 'CANNOT BE SMOOTHED. DO YOU WISH TO <A>BORT'
              PRINT*, 'THE SMOOTHING ROUTINE, OR <E>ENTER NEW'
              PRINT*, 'INPUT DATA? ENTER <A> OR <E>'
              READ*(A),SMOPT
              IF (SMOPT.EQ.'A') THEN
                SMOPT='N'
                GO TO 1855
              ELSEIF (SMOPT.EQ.'E') THEN
                GO TO 2022
              ELSE
                GO TO 2022
              ENDIF
            ELSE
              S1=(H1-H0)/(T1-T0)
              S2=(H2-H1)/(T2-T1)
              ENDIF
              TH=.3*T1 + .2*T0
              THH=.2*T2 + .2*T0
              A0=H0+S1*(TH-T0)
              A1=S1
              A2= H1 + .2*T2*S2 - .2*T1*S2 - A0 - THH*A1
              A2= A2*3 - S2*THH + A1*THH
              A2= A2/THH**2
              A3=(S2 - A1 - 2*A2*THH)/(3*THH**2)
              XDDTH=2*A2+3*A3*THH
              INPPT(K,I,3)=3
              IF (S1.GT.0) AND (A2.LT.0) INPPT(K,I,3)=5

```

```

IF(S1.LT.0.AND.A2.GT.0) INPPT(K,I,8)=5
IF(S2.GT.0.AND.XDDTH.LT.0) INPPT(K,I,8)=5
IF(S2.LT.0.AND.XDDTH.GT.0) INPPT(K,I,8)=5
  IF(INPPT(K,I,8).EQ.5) THEN
    POLYMAT(1,1)=1.
    POLYMAT(1,2)=THH
    POLYMAT(1,3)=THH**2
    POLYMAT(2,1)=1.
    POLYMAT(2,2)=(4./3.)*THH
    POLYMAT(2,3)=(5./3.)*THH**2
    POLYMAT(3,1)=1.
    POLYMAT(3,2)=2.*THH
    POLYMAT(3,3)=(20./6.)*THH**2
    HH=H1+S2*(THH-T1+TH)
    CA2=(HH-A1*THH-A0)/THH**3
    CA3=(S2-A1)/(3*THH**2)
    CA4=0.
    CALL INVERT(POLYMAT,IPOLY,3,3,*1261)
    A2=CA2*IPOLY(1,1)+CA3*IPOLY(1,2)+CA4*IPOLY(1,3)
    A3=CA2*IPOLY(2,1)+CA3*IPOLY(2,2)+CA4*IPOLY(2,3)
    A4=CA2*IPOLY(3,1)+CA3*IPOLY(3,2)+CA4*IPOLY(3,3)
    XTDI=6.*A2+24.*A3*THH+60.*A4*THH**2
    INPPT(K,I,8)=5
    IF((S2-S1).LT.0.AND.A2.GT.0) INPPT(K,I,3)=1
    IF((S2-S1).GT.0.AND.A2.LT.0) INPPT(K,I,3)=1
    IF(A2.GT.0.AND.XTDI.GT.0) INPPT(K,I,8)=1
    IF(A2.LT.0.AND.XTDI.LT.0) INPPT(K,I,8)=1
    IF(INPPT(K,I,3).EQ.5) THEN
      INPPT(K,I,3)=A0
      INPPT(K,I,4)=A1
      INPPT(K,I,5)=A2
      INPPT(K,I,6)=A3
      INPPT(K,I,7)=A4
      GO TO 1262
    ELSE
      GO TO 1262
    ENDIF
1261  INPPT(K,I,3)=0
      PRINT*, 'UNABLE TO SMOOTH INPUT ',K,' AT PT. ',I
      INPPT(K,I,3)=A0
      INPPT(K,I,4)=A1
1262  IF (INPPT(K,I,8).EQ.1) THEN
      F=.01
1263  FI=0
      HH=H1+S2*(THH-T1+TH)
      CA2=(HH-A1*THH/F-A0)/(THH/F)**3
      CA3=(S2-A1)/(3*(THH/F)**2)
      CA4=0.
      POLYMAT(1,1)=1.

```

```

POLYMAT(1,2)=THH/F
POLYMAT(1,3)=(THH/F)**2
POLYMAT(2,1)=1.
POLYMAT(2,2)=(4./3.)*THH/F
POLYMAT(2,3)=(5./3.)*(THH/F)**2
POLYMAT(3,1)=1.
POLYMAT(3,2)=2.*THH/F
POLYMAT(3,3)=(10./3.)*(THH/F)**2
CALL INVERT(POLYMAT,IPOLY,3,3,*1261)
A2=CA2*IPOLY(1,1)+CA3*IPOLY(1,2)+CA4*IPOLY(1,3)
A3=CA2*IPOLY(2,1)+CA3*IPOLY(2,2)+CA4*IPOLY(2,3)
A4=CA2*IPOLY(3,1)+CA3*IPOLY(3,2)+CA4*IPOLY(3,3)
XTDT=6.*A2+24.*A3*(THH/F)+60.*A4*(THH/F)**2
IF ((S2-S1).LT.0.AND.A2.GT.0) FI=1
IF ((S2-S1).GT.0.AND.A2.LT.0) FI=1
IF (A2.GT.0.AND.XTDT.GT.0) FI=1
IF (A2.LT.0.AND.XTDT.LT.0) FI=1
IF (FI.EQ.1) THEN
    F=F*1.02
    GO TO 1263
ELSE
    INPPT(K,I,3)=A0
    INPPT(K,I,4)=A1
    INPPT(K,I,5)=A2
    INPPT(K,I,6)=A3
    INPPT(K,I,7)=A4
    INPPT(K,I,8)=F
ENDIF
ENDIF
ELSE
    INPPT(K,I,3)=A0
    INPPT(K,I,4)=A1
    INPPT(K,I,5)=A2
    INPPT(K,I,6)=A3
    INPPT(K,I,7)=0.
    INPPT(K,I,8)=3
ENDIF
1854    CONTINUE
1855    CONTINUE
ENDIF
ENDIF
IFLAG(22)=1
GO TO 8007

```

The following code is located in PROGRAM OPT26.

```

DO 1861 K=1,P
C*****
C      THIS IS THE CODE WHERE THE INPUT IS GENERATED WHEN      *
C      A CUSTOM INPUT HAS BEEN SELECTED. A0,A1,A2,A3,A4 ARE      *
C      COEFFICIENTS OF THE THIRD OR FIFTH ORDER POLYNOMIAL USED  *
C      TO SMOOTH THE CURVE. T0,T1,H0,H1 ARE THE TIMES AND        *
C      MAGNITUDES AT THE BEGINNING AND END OF THE LINE SEGMENT  *
C      RESPECTIVELY AND S1 (=A1) IS THE SLOPE OF THE LINE.      *
C      THE VALUE CONTAINED IN INPPT(K,I,8) DETERMINES WHETHER  *
C      THE CURVE IS SMOOTHED BY A THIRD ORDER POLY, FIFTH ORDER *
C      POLY, OR A TIME SCALED FIFTH ORDER POLY. OPTION #22      *
C      CONTAINS A DETAILED DESCRIPTION OF THE SMOOTHING METHODS *
C      AND THE DEFINITION OF THE SMOOTHING VARIABLES IN A COMMENT*
C      STATEMENT PRIOR TO THE SMOOTHING ALGORITHM.              *
C*****
      IF (SMOFT.EQ.'N') THEN
        I=1
1401      IF (I.LT.10) THEN
            IF (T.GT.(INPPT(K,I,1))) THEN
                I=I+1
                GO TO 1401
            ENDIF
            ELSE
                I=10
            ENDIF
            IF (I.EQ.1) THEN
                IF (INPPT(K,I,1).EQ.0) THEN
                    V(K)=INPPT(K,I,2)
                ELSE
                    S1=(INPPT(K,I,2)/INPPT(K,I,1))
                    V(K)=S1*T
                ENDIF
            ELSE
                IF ((INPPT(K,I,1)).EQ.(INPPT(K,I-1,1))) THEN
                    V(K)=INPPT(K,I,2)
                ELSE
                    S1=(INPPT(K,I,2)-INPPT(K,I-1,2))
                    S1=S1/(INPPT(K,I,1)-INPPT(K,I-1,1))
                    V(K)=INPPT(K,I-1,2)+S1*(T-INPPT(K,I-1,1))
                ENDIF
            ENDIF
        ENDIF

```

```

ELSE
  I=1
1402 IF (I.LT.10) THEN
      IF (T.GE. (.8*INPPT(K,I,1)+.2*INPPT(K,I+1,1))) THEN
        I=I+1
        GO TO 1402
      ENDIF
    ELSE
      I=10
    ENDIF
    IF (I.EQ.1) THEN
      T0=0
      H0=0
    ELSE
      T0=INPPT(K,I-1,1)
      H0=INPPT(K,I-1,2)
    ENDIF
    T1=INPPT(K,I,1)
    TH=.8*T1+.2*T0
    H1=INPPT(K,I,2)
    A0=INPPT(K,I,3)
    A1=INPPT(K,I,4)
    A2=INPPT(K,I,5)
    A3=INPPT(K,I,6)
    A4=INPPT(K,I,7)
    IF (T.GT.TH.AND.I.LE.9) THEN
      IF (INPPT(K,I,3).EQ.0) THEN
        V(K)=A0+A1*(T-TH)+A2*(T-TH)**2+A3*(T-TH)**3
      ELSEIF (INPPT(K,I,3).EQ.3) THEN
        V(K)=A0+A1*(T-TH)+A2*(T-TH)**3+A3*(T-TH)**4
        +A4*(T-TH)**5
      ELSEIF (T.GE.T1.AND.INPPT(K,I,8).EQ.0) THEN
        S2=INPPT(K,I-1,2)-INPPT(K,I,2)
        S2=S2/(INPPT(K,I+1,1)-INPPT(K,I,1))
        V(K)=H1 + S2*(T-T1)
      ELSEIF (INPPT(K,I,8).NE.0) THEN
        F=1./INPPT(K,I,3)
        V(K)=A0+A1*(T-TH)+F*A2*((T-TH)+F)**3
        +A3*((T-TH)+F)**4+A4*((T-TH)+F)**5
      ENDIF
    ELSE
      V(K)=H0+A1*(T-T0)
    ENDIF
  ENDIF
1861 CONTINUE
ENDIF

```

C. Suppression of Actuators and Sensors

1. Description. It is sometimes convenient to eliminate actuator and or sensor dynamics from the simulation. Previously, this would involve destroying the actuator and sensor data and then re-entering the same data when the dynamics are desired. The actuator and sensor data is now stored in permanent variable locations, while temporary variables can be suppressed and reset in option 4 or option 5.

2. User's Guide. The interactive prompts are self-explanatory for this change and are listed below.

OPTION, PLEASE > #
? 4

THIS OPTION ENTERS THE ACTUATOR STATE EQUATION DATA

ENTER "0" TO SUPPRESS ACTUATORS
ENTER "1" TO SET ACTUATOR VALUES...>
ENTER "2" TO USE STORED ACTUATOR VALUES...)
? 0

OPTION, PLEASE > #
? 5

THIS OPTION ENTERS THE SENSOR STATE EQUATION DATA

ENTER "0" TO SUPPRESS SENSORS
ENTER "1" TO SET SENSOR VALUES...>
ENTER "2" TO USE STORED SENSOR VALUES...)
? 0

It is important to note that if the actuator or sensor dynamics are suppressed, they will not be saved in MEMO when exiting the program. A warning message to this effect has been added to the exit routine and is shown in the "Saving Memory Files Without Exiting" section of this appendix.

3. Programmer's Guide. The code to accomplish this option is very simple but is spread out in options 4, 5, 9, 99, 104, and 105. For these two reasons it is not repeated here. The following variables are added for this feature:

<u>Variable</u>	<u>Description</u>
PNA(I)	Vector of "m" integers (m = number of inputs), each being the number of states in the actuator for that input. This variable is a permanent storage location for the vector variable NA(I), the quantity used by the simulation for actuator state data. NA(I) is set to zero when the actuators are suppressed and is set equal to PNA(I) when the actuators are reset.
PNS(I)	Vector analagous to PNA(I) containing the number of states for each output sensor. NS(I) is the local variable used by the simulation.

D. Saving Memory Files Without Exit

1. Description. Upon selection of option 99, MULTI will save all pertinent data in local files MEM0, MEM10, MEM20, and MEM30 and then the program will either return to normal execution or exit according to the user's desires.

2. User's Guide. Option 99 allows graceful termination of MULTI and automatically saves all plant, actuator, sensor, design and simulation data in local files prior to exiting the program. However, as all MULTI users will inevitably discover, there are a number of ways to exit MULTI involuntarily, leaving the user with the irritating task of re-entering all data that had not been saved. The most commonly encountered inadvertent termination of MULTI occurs when the user enters a "RETURN" at the prompt without any data preceding the "RETURN". The computer program has interrupted execution at a read statement and is expecting input from the terminal. If no input is provided, an "END OF FILE" is encountered and the program aborts execution. Naturally, this phenomenon is accompanied by the loss of all volatile data, which may have taken hours to generate. This problem has not been corrected, but if the user is cautious to save data regularly, the frustration of re-entering data can be avoided and one is likely to stay motivated toward the ultimate objective for a considerably longer time. The

procedure for saving data is quite simple as is demonstrated below:

```
OPTION, PLEASE > #  
? 99
```

```
ALL PLANT INPUT DATA HAS BEEN SAVED IN A LOCAL FILE  
    CALLED "MEM0"
```

```
ALL DESIGN DATA HAS BEEN SAVED IN A LOCAL FILE  
    CALLED "MEM10"
```

```
ALL SIMULATION DATA HAS BEEN SAVED IN A LOCAL FILE  
    CALLED "MEM20"
```

```
ALL MONTE CARLO SIMULATION DATA HAS BEEN SAVED  
IN A LOCAL FILE CALLED "MEM30"
```

```
*****  
* ACTUATORS AND SENSORS WERE NOT SAVED *  
*****
```

```
DO YOU WANT TO EXIT MULTI: Y OR N ?  
? N
```

```
OPTION, PLEASE > #  
?
```

3. Programmer's Guide. The code changes required to accomplish the desired changes to option 99 consist of several conditionals to determine whether sensors and actuators have been suppressed, generation of a warning based on that determination and finally a question asking the user if termination is desired. The only variable introduced is a logical character "EXIT", depending on the user's desires. The exit routine code reads as follows:

```

C -----ROUTING FOR OPTION #99-----
  ELSEIF (NOPT.EQ.99) THEN
    IF (IPLOT.GT.0) THEN
      CALL PLOTE (BLK)
      PRINT '(A/)', ' REMINDER: ROUTE ''PLOT(S)'' BEFORE LOGOUT!'
    ENDIF
    CALL OVERLAY (MULTI,12,0)
    IF (ACT.EQ.'N'.OR.SEN.EQ.'N') THEN
      PRINT*, '*****'
      IF (ACT.EQ.'Y') THEN
        PRINT*, '* NOTE: SENSOR DYNAMICS WERE NOT SAVED *'
      ELSEIF (SEN.EQ.'Y') THEN
        PRINT*, '* NOTE: ACTUATOR DYNAMICS WERE NOT SAVED *'
      ELSE
        PRINT*, '* ACTUATORS AND SENSORS WERE NOT SAVED *'
      ENDIF
      PRINT*, '*****'
    ENDIF
    PRINT*, 'DO YOU WANT TO EXIT MULTI: Y OR N ?'
    READ '(A)', EXIT
    IF (EXIT.EQ.'N') THEN
      GO TO 9000
    ENDIF
    PRINT '(A/)', ' HAVE A NICE DAY!'
    STOP

```

E. Convert Input Vector "U"
From Radians to Degrees

1. Description. After completion of option 26 the user is given the option of converting several of the data arrays from radians to degrees prior to plotting the data. Previously this option did not include the control input vector "U". The routine now includes this conversion as well, to account for plants in which the input matrix is given in terms of radians. The original code is the work of Major Terry L. Courtheyn (6:C-1). Courtheyn's work is merely copied to accomplish the additional conversion.

2. User's Guide. The prompts for this option are identical to the prompts originally programmed by Courtheyn with the addition of a similar prompt for the conversion of the "U" vector. Both the use and programming of this change are self-explanatory and the programmer's guide is omitted.

F. Plot Combination of States and Inputs

1. Description. Often it is desired to plot not only a state but its derivative as well. In the case of an aircraft, it is often convenient to plot the normal acceleration as a function of time, requiring a combination of states and state derivatives. Since in a linear system of equations any state derivative can be described in terms of the states and inputs, all that is required is to be able to combine state and input data to obtain any function of states and state derivatives as a function of time. This change expands the existing capability of plotting combinations of states to the option of plotting a user definable combination of states and inputs.

2. User's Guide. Following is the interactive dialog that the user will encounter after selecting one of the six plotting options (31-36). This particular example is a terminal plot option (31). To obtain a plot of some combination of states and inputs, the user selects plot choice "4" at the prompt.

THIS OPTION PRODUCES A PLOT AT YOUR TERMINAL

PLEASE CHOOSE ONE OF THE FOLLOWING:

FOR A SINGLE SAMPLING TIME

- 1...A PLOT OF UP TO 2 INPUT AND OUTPUT PAIRS
- 2...A PLOT OF UP TO 4 INPUTS OR OUTPUTS OR STATES
- 3...A PLOT OF UP TO 4 DIFFERENT SIMULATIONS
(FOR ANY SINGLE INPUT OR OUTPUT)

OR 4...A PLOT OF UP TO 4 COMBINATIONS OF STATES

ENTER CHOICE DESIRED >

? 4

CHOICE #4...YOU'VE CHOSEN TO PLOT COMBINATION OF STATES

ENTER THE NUMBER OF COMBINATIONS
OF STATES AND INPUTS.....>

? 1

ENTER "Z" MATRIX...1 ROWS WITH 4 ELEMENTS EACH
ROW 1 >

? 1,2,3,4

COMBO MATRIX Z...

.1000E+01 .2000E+01 .3000E+01 .4000E+01

IS THIS CORRECT...YES,NO,\$...>

? Y

Up to this point, the user sees no change in the interactive prompts. Now the program requires entry of the inputs to be included in the combination. As with states, the user enters the matrix which adds the weighted inputs into the desired combination.

DOES THE COMBO INCLUDE INPUTS?.....Y OR N
 ? Y

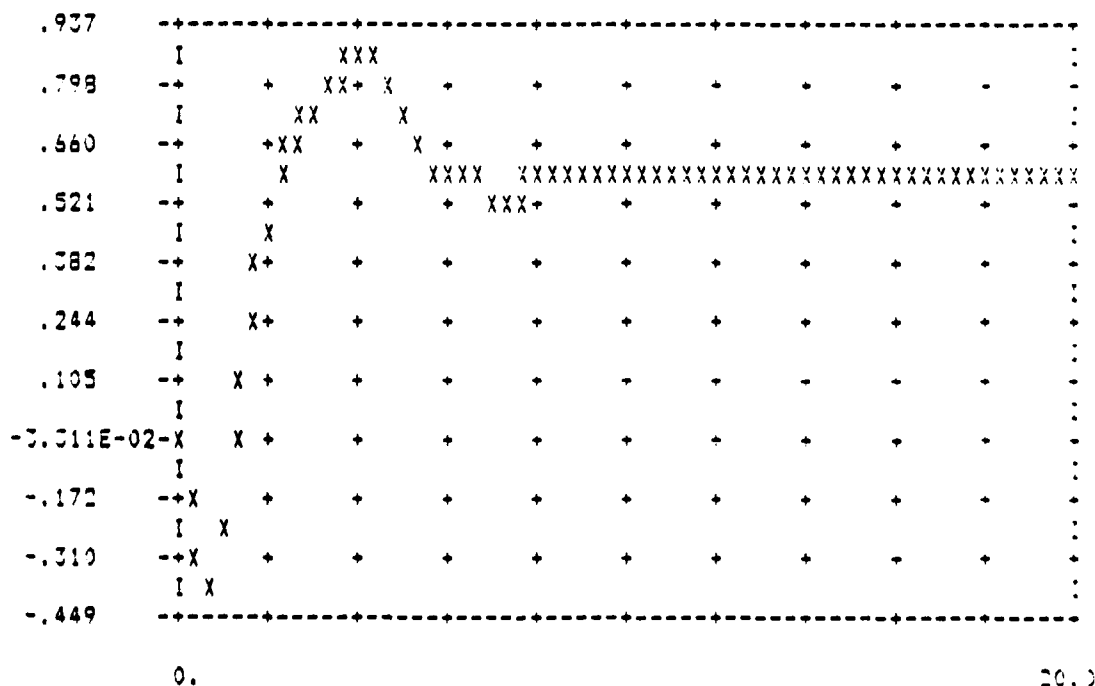
ENTER "ZU" MATRIX... 1 ROWS WITH 3 ELEMENTS
 ROW 1 >
 ? 1,2,3

COMBO MATRIX ZU...

.1000E+01 .2000E+01 .3000E+01

IS THIS CORRECT...YES,NO,\$...>
 ? Y

FOR NO GRID ON PLOT ENTER "0", FOR A GRID ENTER "1" >
 ? 1



CURVE X ABOVE IS COMBO 1

DO YOU WANT A LIST OF POINTS USED IN PLOTTING?

ENTER...YES OR NO...>

? N

OPTION, PLEASE > #

?

3. Programmer's Guide. All of the code to produce plots, either at the terminal or files for CALCOMP plotting, is located in three overlays: OPTPLOT, OPT31, and OPT34. Although there are six options (31-36) that require the combining of states and inputs for plotting, there is only one routine to accomplish the calculations and it is located in OPTPLOT. The code to combine the inputs is nearly identical to the code to combine the states that was originally in MULTI. This code, listed below, requires the addition of two variables.

<u>Variable</u>	<u>Description</u>
ZU(I,J)	Array containing the coefficients used to combine (I) inputs into (J) combinations.
IINP	Logical character indicating the presence of inputs in the combination to be plotted.

```

1486 PRINT '(/A)', ' ENTER THE NUMBER OF COMBINATIONS'
      PRINT*, 'OF STATES & INPUTS.....>'
      READ*, K
      DO 1499 I=1,K
      DO 1497 J=1,M
1497 Z(I,J)=0
      DO 1498 L=1,INPS
1498 ZU(I,L)=0
1499 CONTINUE
      PRINT*,
      PRINT*, 'ENTER "Z" MATRIX... ',K, ' ROWS WITH ',M, ' ELEMENTS EACH'
      DO 1490 I=1,K
      PRINT*, 'ROW ',I, ' >'
1490 READ*, (Z(I,J),J=1,M)
      PRINT*,
      GO TO 190
1495 CALL FIX (Z,K,M)
190 PRINT*, 'COMBO MATRIX Z...'
      CALL MATPR (Z,K,M)
      CALL ANSWER (*1495,*8010)
      PRINT*,
      PRINT*, ' DOES THE COMBO INCLUDE INPUTS?.....Y OR N'
      READ '(A)',IINP
      IF (IINP.EQ.'Y') THEN
        GO TO 1502
      ELSE
        GO TO 1503
      ENDIF
C *****
C ***** ZU MATRIX FORMS THE COMBO OF INPUTS *****
C *****
1502 PRINT*,
      PRINT*, ' ENTER "ZU" MATRIX... ',K, ' ROWS WITH ',INPS, ' ELEMENTS'
      DO 1491 I=1,K
      PRINT*, 'ROW ',I, ' >'
1491 READ*, (ZU(I,J),J=1,INPS)
      PRINT*,
      GO TO 191
1492 CALL FIX (ZU,K,INPS)
191 PRINT*, ' COMBO MATRIX ZU...'
      CALL MATPR (ZU,K,INPS)
      CALL ANSWER (*1492,*8010)
1503 DO 1510 I=1,N
      DO 1505 J=1,K
      S=1.
      DO 1500 L=1,INPS
1500 S = S + U(I,L+1)*ZU(J,L)
      DO 1501 L=1,M

```

```

1501 S = S + X(I,L+1)*Z(J,L)
1505 PLMAT(I,J)=S
1510 CONTINUE
      ICLM=K
      CHOICE=' COMBO'

```

```

C -----
1515 IF (IFLTEN.EQ.0) IFLTEN=1
      IFLTEN=IFLTEN-1
      IF (IFLTEN.NE.0) GO TO 1520
      IF (ICODE.EQ.3) ICLM=LINES
      GO TO 9010

```

G. Simulation of Nonlinearities
Peculiar to Aircraft

1. Description. Linear models of aircraft are usually quite accurate, provided, of course, the assumptions made in obtaining the linear model are not violated in the simulation. One of the key assumptions is that the forces and moments on the aircraft are linear with control surface deflection. If control surface deflections are large, as in maximum performance maneuvers, or in the case of inherently nonlinear control surfaces like vectored, variable thrust, a linear model is inadequate. For the reasons described in Chapter III the principal nonlinearity of large longitudinal control surface deflections is the reversal of the sign of the partial derivative of velocity with respect to the deflection when the surface passes through zero angle of attack. This phenomenon is easily modeled in the simulation and is implemented in a special version of MULTI customized for aircraft models. In the case of vectored, variable thrust and nonlinearities are

more complex. A rigorous development of the nonlinear effects of this type of input is contained in Chapter IV. These effects are also simulated, at the user's option, in the customized version of MULTI.

2. User's Guide. Unfortunately, to implement these two features it is necessary to place additional requirements on the allowable form of the model used in the customized version of MULTI. These constraints are as follows:

a. The plant must be longitudinal, body axis, linearized model of an aircraft.

b. The states are defined as THETA (pitch angle), U (X-axis velocity), Q (pitch rate), and ALPHA (angle of attack). These are all perturbation values and must be arranged in that order. Additional states are allowed but must be after ALPHA in the state vector.

c. The first two inputs must be aerodynamic surfaces, like stabilators or canards. The third input must be a variable thrust input like a throttle or reverser vanes. If a two-dimensional nozzle is desired, its deflection angle must be the fourth input.

d. The equilibrium angle of attack for each of the aerodynamic surfaces, and the equilibrium deflection of the two-dimensional nozzle must be known and entered in option #3.

It is important to note that when a two-dimensional nozzle is used, not only is the simulation nonlinear, but the solution is not unique. This of course means that there are an infinite number of steady state solutions and it may be difficult to obtain one that is satisfactory. It is left to the user to determine how one finds a satisfactory solution. A sample of the interactive prompts for option #3 follows:

```

OPTION, PLEASE > #
? N
  ENTER EQUILIBRIUM VALUE FOR EACH INPUT
    INPUT 1
    ? -.01
    INPUT 2
    ? .2
    INPUT 3
    ? .3
  IS THERE A TWO-DIMENSIONAL NOZZLE INPUT ON THIS
  AIRCRAFT? (Y OR N)
  ? Y
    ENTER THE NOZZLE MOMENT ARM FROM CG (FT) >
    ? 26
    ENTER PITCH MOMENT OF INERTIA IYY (SLUG*FT**2) >
    ? 1265489
    ENTER THE AIRCRAFT MASS (SLUGS) >
    ? 1437.2
    ENTER THE EQUILIBRIUM VELOCITY (FT/SEC) >
    ? 201.3
    ENTER THE DERIVATIVE Z-ALPHA-DOT (FT/SEC) >
    ? 1.256

OPTION, PLEASE > #
?

```

3. Programmer's Guide. There are two blocks of code added to MULTI for this feature. The first block is

located in PROGRAM OPT0 under option #3 and is the interactive portion where the user enters the necessary data. The second block of code is located in PROGRAM OPT26 and is where the nonlinearities are actually computed during the simulation.

<u>Variables</u>	<u>Description</u>
EV(I)	Vector containing the equilibrium angles of attack for aerodynamic surfaces and initial nozzle and thrust input values.
EVA(I)	Vector containing time varying angles of attack for the aerodynamic surfaces.
NOZ2D	Logical character indicating presence of a two-dimensional nozzle.
LX	Local real variable, nozzle moment arm.
IYY	Local real variable, aircraft pitch moment of inertia.
MASS	Local real variable, aircraft mass.
UEQ	Local real variable, equilibrium velocity.
ZAD	Local real variable, body axis coefficient of force in the z direction with respect to the time derivative of the angle of attack.
BNOZ1	Real variable, nonlinear input matrix coefficient.
BNOZ2	Real variable, nonlinear input matrix coefficient.

BNL(K,J) Array containing original input matrix plus
 the nonlinear effects of BNOZ1, BNOZ2, and
 sign change.

The following code is located in PROGRAM OPTO:

```
PRINT*, 'ENTER EQUILIBRIUM VALUE FOR EACH INPUT'
DO 311 I=1,M
PRINT*, 'INPUT ',I
READ*, EV(I)
311 CONTINUE
PRINT*, 'IS THERE A TWO-DIMENSIONAL NOZZLE INPUT ON THIS'
PRINT*, 'AIRCRAFT? (Y OR N)'
READ '(A)',NOZ2D
IF (NOZ2D.EQ.'Y') THEN
PRINT*, 'ENTER THE NOZZLE MOMENT ARM FROM CG (FT)>'
READ*, LX
PRINT*, 'ENTER PITCH MOMENT OF INERTIA IYY (SLUG*FT**2)>'
READ*, IYY
BNOZ1=LX/IYY
PRINT*, 'ENTER THE AIRCRAFT MASS (SLUGS) >'
READ*, MASS
PRINT*, 'ENTER THE EQUILIBRIUM VELOCITY (FT/SEC) >'
READ*, UEQ
PRINT*, 'ENTER THE DERIVATIVE Z-ALPHA-DOT (FT/SEC) >'
READ*, ZAD
BNOZ2=1./(MASS*(UEQ-ZAD))
ELSE
NOZ2D='N'
ENDIF
IFLAG(3)=1
IPLANT=1
GO TO 9001
```

The remaining code for this feature is located in PROGRAM
OPT26:

```

C*****
C** CODE TO HARDWIRE NONLINEARITIES FOR STOL F-15      ****
C*****
      EVA(1)=EV(1)+X(4)
      EVA(2)=EV(2)+X(4)
      EVA(3)=EV(3)
      IF(NOZ2D.EQ.'Y') EVA(4)=EV(4)
      DO 1204 I=1,N
      BNL(I,3)=B(I,3)
      DO 1203 J=1,2
      BNL(I,J)=B(I,J)
      IF(EVA(J).GE.0) THEN
        IF((U(J) + EVA(J)).LT.0) THEN
          BNL(2,J)=-B(2,J)
        ENDIF
      ELSE
        IF((U(J) + EVA(J)).GT.0) THEN
          BNL(2,J)=-B(2,J)
        ENDIF
      ENDIF
1203 CONTINUE
1204 CONTINUE
      IF(NOZ2D.EQ.'Y') THEN
        BNL(3,3)=B(3,3)+BNOZ1*X(5)
        BNL(4,3)=B(4,3)+BNOZ2*X(5)
      ENDIF
C*****      END NON-LINEARITIES *****

```

H. Calculate Initial Integrator State Vector ZO

1. Description. MULTI requires two vectors of initial conditions to specify an initial system state. The first is the initial conditions desired on each of the plant states. In the case of an aircraft this specifies the initial orientation and motion of the aircraft. The second vector is the initial conditions imposed by the integral of the error vector $Z(0)$. If initial control

surfaces deflections are desired they must be specified by the $Z(0)$ vector. The relationship between $Z(0)$ and $U(0)$ is given by:

$$Z(0) = g K_1^{-1} U(0)$$

g = forward loop gain (1/SAMPT)

K_1 = controller integral gain matrix

$U(0)$ = initial control surface deflections

Option #6 now includes a routine that calculates the $Z(0)$ vector using the current values of K_1 and g as well as a user specified $U(0)$.

2. User's Guide. This feature is invoked by selecting option 6 and making the appropriate choice from the menu (shown in the example below). The program requests the desired initial control inputs, calculates the $Z(0)$ vector and stores it in the appropriate memory location.

```
THIS OPTION COMPUTES THE TRANSFER FUNCTIONS OF THE SYSTEM

FOR OPEN-LOOP TRANSFER FUNCTION ENTER 1.
FOR CLOSED-LOOP TRANSFER FUNCTION ENTER 2 >
FOR G(0) AND G(0) INVERSE MATRICES ENTER 3 >
TO CALCULATE Z(0) VECTOR ENTER 4
? 4
ENTER THE 3 ELEMENTS OF THE DESIRED U(0) VECTOR
? 2,5,-6
```

```
THE INITIAL CONDITION ZO(1) HAS BEEN SET TO:
624.0088888888 -200.7108585859 49.21010101011
```

```
OPTION. PLEASE > #
```

3. Programmer's Guide. The code for this feature resides in PROGRAM XFERFN, the overlay for computing the system transfer functions.

```

362 PRINT '(A)', ' FOR OPEN-LOOP TRANSFER FUNCTION ENTER 1,'
PRINT*, 'FOR CLOSED-LOOP TRANSFER FUNCTION ENTER 2 > '
PRINT*, 'FOR G(0) AND G(0) INVERSE MATRICES ENTER 3 > '
PRINT*, 'TO CALCULATE Z(0) VECTOR ENTER 4 > '
READ*, Tftype
*****
C* THIS SECTION CALCULATES THE NECCESARY INITIAL CONDITION VECTOR Z(0) *
C* TO SPECIFY A VECTOR OF INITIAL CONTROL INPUTS U(0). THE EQUATION *
C* USED FOR THIS CALCULATION IS: *
C*       $U(0) = G \cdot K1 \cdot Z(0) + G \cdot K0 \cdot ZDOT(0)$  *
C* ASSUMING THAT THESE INITIAL CONDITONS ARE IMPOSED TO ESTABLISH AN *
C* EQUILIBRIUM CONDITION WITH NON-ZERO CONTROL SURFACE DEFLECTIONS, *
C*  $ZDOT(0) = 0$ , AND THE EQUATION REDUCES TO: *
C*       $U(0) = G \cdot K1 \cdot Z(0)$  *
C* *
C*       $Z(0) = (K1 \text{ INVERSE}) \cdot U(0) / G$  *
C*****
IF(Tftype.EQ.4) THEN
    PRINT*, 'ENTER THE 'M,' ELEMENTS OF THE DESIRED U(0) VECTOR'
    READ*, (U(I), I=1,M)
    CALL MATPR(K1,P,P)
    CALL INVERT(K1,K1I,P,PD,*4569)
    DO 4568, I=1,P
        ZO(I)=0.
    DO 4567, J=1,P
        ZO(I)=ZO(I)+K1I(I,J)*U(J)+SAMPT
4567 CONTINUE
4568 CONTINUE
    PRINT*, 'THE INITIAL CONDITION ZO(I) HAS BEEN SET TO:'
    PRINT*, (ZO(I), I=1,P)
    GO TO 4570
4569 PRINT*, 'K1 MATRIX IS SINGULAR AND CANNOT BE INVERTED'
4570 GO TO 8017
ENDIF

```

I. Program Outline

1. Introduction. The intent of this section is to provide a programmer's guide for the entire MULTI program. A copy of this outline can be found on the magnetic tape containing the master copy of MULTI. Additions and revisions by future users is highly encouraged and will eventually result in thorough documentation.

2. List and Description of Major Program Elements. Following is a list and brief description of all of the overlays and subroutines contained in MULTI. The program elements are listed in the order that they occur in the program listing.

a. PROGRAM EXEC. This overlay is the master program for MULTI and organizes its execution. After printing the beginning message, EXEC requests the user's choice of options and routes execution to one of the other seventeen overlays depending on the response. The exit routine is also contained in EXEC.

b. SUBROUTINE MATPR. This subroutine is used to print matrices.

c. SUBROUTINE QPRINT. This subroutine asks the user if a particular set of data should be printed at the terminal.

d. SUBROUTINE ANSWER. After printing data, ANSWER asks if the data is correct as printed.

e. SUBROUTINE INVERT. This subroutine formats a matrix and its associated parameters for inversion by the IMSL routine, LINV2F.

f. SUBROUTINE FIX. If, in SUBROUTINE ANSWER, the user desires to change a matrix, this subroutine accepts the changes and updates the matrix.

g. PROGRAM OPT0. This overlay contains the routines for the plant input options (options 0 through 9). However, option 6 is a separate overlay called XFERFN.

h. PROGRAM OPT10. This overlay contains all the design parameter routines (options 10 through 19) with the exception of options 14 and 18.

i. PROGRAM OPT14U. OPT14U calculates the controller matrices K_0 and K_1 for designs in which the plant parameters are unknown.

j. PROGRAM OPT14R. K_0 and K_1 are calculated in OPT14R for regular plants (first Markov parameter non-zero).

k. PROGRAM OPT14I. Irregular plant controller matrices are calculated in this overlay.

l. PROGRAM OPT18. In the case of an irregular plant, a measurement matrix is required. Option 18 (contained in OPT18) provides several utility routines that can be useful in choosing an appropriate measurement matrix.

m. PROGRAM OPT20. With the exception of options 26 and 28, all simulation options (20 through 29) are contained in OPT20.

n. PROGRAM OPTPLT. OPTPLT is the first of four routines (three overlays and a subroutine) written to generate plots. OPTPLT is the interactive portion in which the user selects the type of plot and the necessary parameters (options 30 through 39).

o. PROGRAM OPT31. Upon selection of one of the terminal plot options (31-33), OPT31 interactively asks for data specifically required for terminal plots. The data is then formatted for use by the terminal plot subroutine PLOTIT.

p. SUBROUTINE PLOTIT. This subroutine is an adaptation of the generalized routine used to produce plots on the line printer. It produces a plot at the user's terminal using non-graphics characters.

q. PROGRAM OPT34. OPT34 transforms the data for plotting into the form required by the CALCOMP plotting routine.

r. PROGRAM ERROR. This overlay contains all messages that result from errors that are neither fatal nor terminal. These errors are usually a result of attempting to perform calculations requiring data that has not yet been entered.

s. PROGRAM MEMORY. Upon selection of option 99, EXEC routes execution to MEMORY for generation of memory files MEM0, MEM10, MEM20. Section 3 contains the format of the files generated.

t. PROGRAM PRINT. PRINT contains all of the 100 series options that print the current values of the data generated in any of the input options.

u. PROGRAM OPT14B. This overlay computes the controller matrices when the BSTAR method is chosen in option 14.

v. PROGRAM XFERFN. This is the overlay that executes option 6. This option includes computation of any open or closed loop transfer function, steady state transfer functions, and initial controller integrator states.

w. SUBROUTINE PHOFS. This subroutine, called by XFERFN, calculates the transfer function denominator polynomials.

x. SUBROUTINE CADJB. CADJB is also called by XFERFN and computes the transfer function numerator polynomials.

y. SUBROUTINE POLYRT. POLYRT calculates the roots of the polynomials generated by PHOFS and CADJB.

z. SUBROUTINE CLMAT. This subroutine calculates the closed loop matrix used by XFERFN to compute the closed loop transfer functions.

aa. PROGRAM OPT26. OPT26 is the overlay that performs the simulation. It is important to note that the simulation integrates one calculation step at a time, allowing the introduction of noise, nonlinear effects like control surface saturation, and data packing for plotting.

bb. SUBROUTINE CLPASS. CLPASS is the first of four subroutines called by OPT26 to form the differential equations prior to invoking the library routine ODE to solve them. CLPASS is used to form the equations when both actuator and sensor dynamics are present.

cc. SUBROUTINE CLPSS1. CLPSS1 is used to form the differential equations when only actuator dynamics are present.

dd. SUBROUTINE CLPSS2. CLPSS2 is used to form the differential equations when only sensor dynamics are present.

ee. SUBROUTINE CLPSS3. CLPSS3 is used to form the differential equations when neither actuator nor sensor dynamics are present.

ff. SUBROUTINE GPNML. This subroutine uses the IMSL library routine GGNML to produce a zero mean, gaussian random vector with a standard deviation of 1. GPNML uses this normalized random vector to obtain a random vector with the mean and standard deviation required by the various noise inputs.

gg. SUBROUTINE YOUT. This subroutine, called by OPT26, computes the output vector from the state vector and "C" matrix.

hh. PROGRAM OPT28. OPT28 executes the figures of merit calculations of option 28. The figures of merit are based solely on the empirical data calculated during the simulation. No theoretical techniques such as the Laplace final value theorem have been implemented. Option 28 can be executed only once for each simulation, after which all figure of merit data is inaccessible.

3. Memory Files. MULTI generates four local memory files to prevent the user from having to enter all the required data for each execution. These files have a specific format that must be maintained if the user chooses to manually create or edit the data. The user should be aware that any file he intends to use must be a local file prior to entry into the MULTI program. Following are examples of each of the data files.

a. MEM0. This file contains the plant, actuator and sensor data.

1	!-led for plant data present
4 3 3	!-states, inputs, outputs
0. 0. 1. 0.	:
-31.5441 -0.0690991 -40.1609 .33519	: A Matrix
0. -.000760356 -.99125 1.36688	:
-.00232295 .00000922542 .979631 -.639215	:
0. 0. 0.	:
-1.1142 -3.1378 -11.79	: B Matrix
.371 -1.57337 -.02507	:
-.05209 -.0766 0.	:

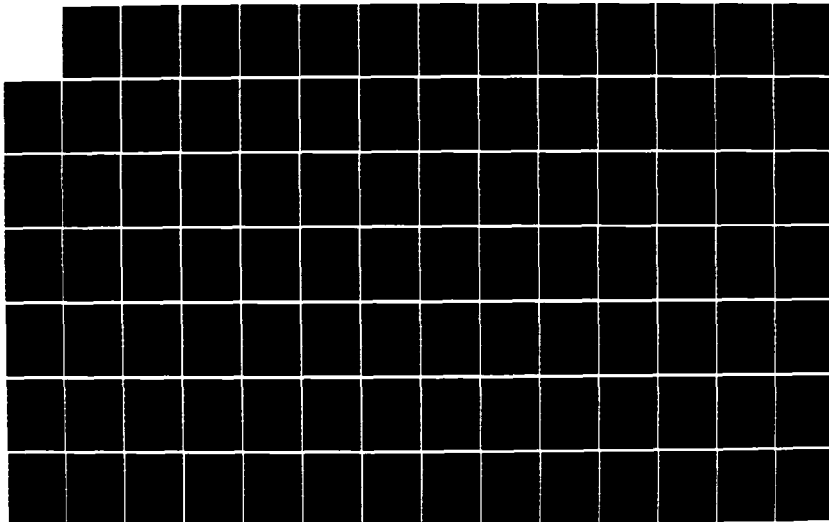
AD-A164 017

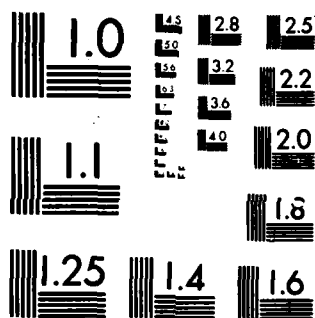
MULTIVARIABLE CONTROL LAW DESIGN FOR ENHANCED AIR
COMBAT MANEUVERING F-15. (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. K A SHEEHAN
DEC 85 AFIT/GE/EE/85D-38 F/G 1/2

4/5

UNCLASSIFIED

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

0. 1. 0. 0.	:
0. 0. 0. 1.	: C Matrix
1. 0. 0. -1.	:
Y	: Is there a D Matrix?
1. 2. 3.	:
2. 3. 1.	: D Matrix
3. 1. 2.	:
1. 2.5. 3.	: Equilibrium surface position **
Y	: Is there a 2-d nozzle? **
1.231 .345	: Nozzle effect on B Matrix **
Y	: Are there actuator dynamics?
2. 2. 1.	: #states in each actuator
0. 1.	: Actuator 1: A Matrix
-8356.198 -303.32	:
0. 8356.198	: Actuator 1: B Matrix
1. 0.	: Actuator 1: C Matrix
0. 1.	:
-8356.198 -303.32	: Actuator 2: A, B,
0. 8356.198	: and C Matrices
1.25 0.	:
-39.	:
39.	: Actuator 3: A, B,
1.	: and C Matrices
Y	: Are there sensor dynamics?
2. 2. 2.	: #states in each sensor
0. 1.	: Sensor 1: A Matrix
-1200. -70.	:
0. 1200.	: Sensor 1: B matrix
1. 0.	: Sensor 1: C Matrix
0. 1.	:
-1926. -223.	: Sensor 2: A, B,
0. 1926.	: and C Matrices
1. 0.	:
0. 1.	:
-1120. -34.	: Sensor 3: A, B,
0. 1120.	: and C Matrices
1. 0.	:

** These entries are found only in MEMO files intended for use in the MULTI version designed for aircraft.

b. MEM10. This file contains the design data.

```

I                                     : Type of design (R,I,U,B)
.01                                  : Alpha
15. .4 .4                           : Sigma matrix diagonal elements
.05                                  : Epsilon
-.000444086307 .0411717192 .252760573 :
.000301990284 -.0279978440 .0892128160 : K0 Matrix
-.0344416265 .00205523127 -.0261815938 :
-.0000044408630 .0004117171 .0025276057 :
.0000030199028 -.0002799784 .0008921281 : K1 Matrix
-.00034441626 .00002055231 -.0002618159 :
0.                                   :
.25                                  : Measurement Matrix
0.                                   :
0. 1. 0. 0.                         :
0. 0. .25 1.                        : F Matrix
1. 0. 0. -1.                        :

```

c. MEM20. This file contains the simulation data.

```

0. 0. 0. 0.                         : Initial state vector (0)
0. 0. 0.                             : Initial integrator vector Z(0)
S                                     : Custom or Standard input?
0. 0. 0. 0.                         :
0. 0.                                 :
0. 0. 0. 0.                         : Data for standard input
0. 0.                                 :
.8 -.1047 25. 25.                   :
1. 0.                                 :
-1.E+10 1.E+10                      :
-1.E+10 1.E+10                      : Control surface limits
-1.E+10 1.E+10                      :
.025                                  : Sample Time
20.                                   : Total simulation time
.025                                  : Calculation step size
1.                                    : Computational delay
0. 0. 0.                             : Output noise means
0. 0. 0.                             : Output noise standard deviation
0. 0. 0. 0.                         : Disturbance noise means
0. 0. 0. 0.                         : Disturbance noise deviations
0. 0. 0.                             : Measurement noise means

```

0. 0. 0.	_:	Measurement noise deviations
0. 0. 0. 0.	:	
0. 0. 0. 0.	:	Disturbance Noise
0. 0. 0. 0.	:	G Matrix
0. 0. 0. 0.	_:	
1 1 1	:	Noise flags

d. MEM30. This file contains the data used in the Monte Carlo noise simulation. It is recommended that the user not tamper with this file since it contains a great deal of raw plot data with little apparent meaning. Thus, an example is not shown here.

Appendix B: Multivariable Control Theory

(Edited and reproduced from Reference 2)

This thesis uses the multivariable design method of Professor Brian Porter of the University of Salford, England (18). The design method employs output feedback with high-gain error-actuated controllers. Output feedback is advantageous since state variables may be difficult to measure while system response data are more readily available.

System State Equations

Porter's method works equally well for either continuous or discrete systems, but it is often easier to first examine a system in the continuous time domain. This is because of the numerical accuracy problem with designing in the z-plane. A continuous time system is represented by the state space model:

$$\begin{aligned}\dot{\underline{x}} &= \underline{A}\underline{x} + \underline{B}\underline{u} \\ \underline{y} &= \underline{C}\underline{x}\end{aligned}\tag{B-1}$$

where

\underline{A} = continuous plant matrix ($n \times n$)

\underline{B} = continuous input control matrix ($n \times m$)

\underline{C} = continuous output matrix ($l \times n$)

\underline{x} = state variable vector with n states

\underline{u} = input vector with m inputs

\underline{y} = output vector with l outputs

The system inputs for an aircraft are the control deflections or actuator input commands, and the system outputs are aircraft responses affected by the inputs.

The method does not allow for a feedforward, \underline{D} , matrix. If such a matrix is present in the original state space model, the control inputs must be redefined as states so that the \underline{D} matrix is absorbed into the \underline{C} matrix. This can be accomplished by incorporating the actuator dynamics into the plant model. Actuator inputs then become control inputs.

To employ Porter's method, it is desirable (but not necessary) to partition the system state equations as follows:

$$\begin{bmatrix} \dot{\underline{x}}_1 \\ \dot{\underline{x}}_2 \end{bmatrix} = \begin{bmatrix} \underline{A}_{11} & \underline{A}_{12} \\ \underline{A}_{21} & \underline{A}_{22} \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} + \begin{bmatrix} \underline{B}_1 \\ \underline{B}_2 \end{bmatrix} \underline{u} \quad (\text{B-2a})$$

$$\underline{y} = \begin{bmatrix} \underline{C}_1 & \underline{C}_2 \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} \quad (\text{B-2b})$$

The equations are partitioned so that \underline{B}_2 and \underline{C}_2 are square ($m \times m$) and ($l \times l$) matrices, respectively. The method requires that the number of inputs to the system equals

the number of outputs which means $m = \ell$, and therefore the dimension of \underline{B}_2 equals the dimension of \underline{C}_2 . It is always possible to form the state equations so that $\underline{B}_1 = \underline{0}$. Sometimes, however, a transformation matrix \underline{T} is necessary to achieve $[0, \underline{B}_2]$ form. In this case, the transformed states no longer have the same physical significance that the original states once had.

For the discrete case the system equations are written as follows:

$$\begin{aligned}\underline{x}[(k+1)T] &= \underline{p}\underline{x}(kT) + \underline{\psi}u(kT) \\ \underline{y}(kT) &= \underline{\Gamma}\underline{x}(kT)\end{aligned}\tag{B-3}$$

where

$\underline{p} = \exp(\underline{A}T)$ = discrete plant matrix

$\underline{\psi} = \int_0^T \exp(\underline{A}T)\underline{B}dt$ = discrete input control matrix

$\underline{\Gamma} = \underline{C}$ = discrete output matrix

In the above equations T is the sampling period, and k takes on integer values from zero to plus infinity.

System With Output Feedback

Figure B.1 shows the block diagram for a continuous output feedback system, where \underline{y} is the command input vector, and \underline{y} is the desired output vector. The blocks for the plant are derived directly from the system state equations,

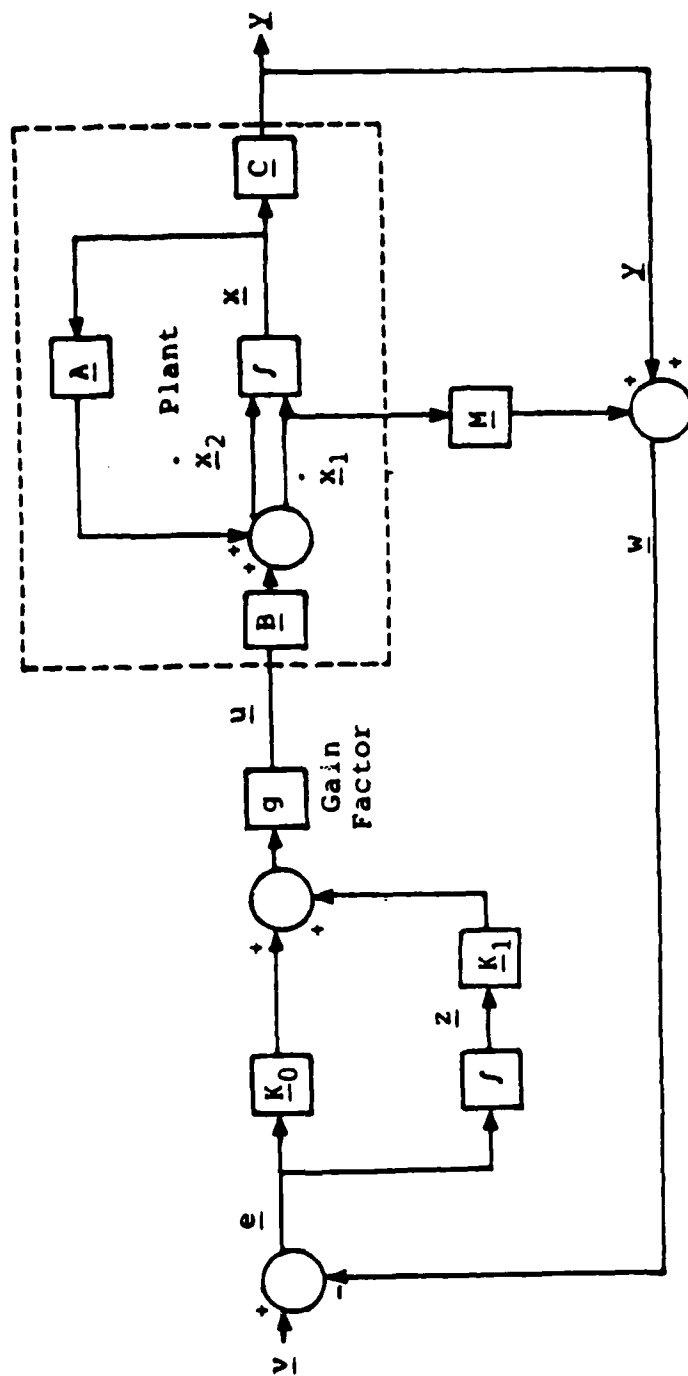


Fig. B.1. System Block Diagram--Continuous Case

Equation (B-1). The proportional plus integral controller has three parameters, \underline{K}_0 , \underline{K}_1 , and g , which must be determined by the designer. The output signal of the controller, \underline{u} , is given in the following control law equation:

$$\underline{u} = g(\underline{K}_0 \underline{e} + \underline{K}_1 \int \underline{e} dt) \quad (B-4)$$

where

\underline{u} is the output signal of the controller

\underline{e} is the error signal at the input of the controller

\underline{K}_0 is the proportional gain matrix

\underline{K}_1 is the gain matrix for the integral term

g is the scalar forward path gain

Figure B.1 is the depiction of a system with only first-order integration in the controller design. The theory allows for a q -dimensional bank of integrators in which case the controller is made up of $(q + 1)$ \underline{K} matrices, \underline{K}_0 thru \underline{K}_q . A measurement matrix \underline{M} is included in the system if the plant is irregular. Regular and irregular plants are discussed later.

The discrete system block diagram, shown in Figure B.2, is similar to the continuous system, but Equation (B-4) becomes

$$\underline{u}(kT) = (1/T) [\underline{K}_0 \underline{e}(kT) + \underline{K}_1 \underline{z}(kT)] \quad (B-5)$$

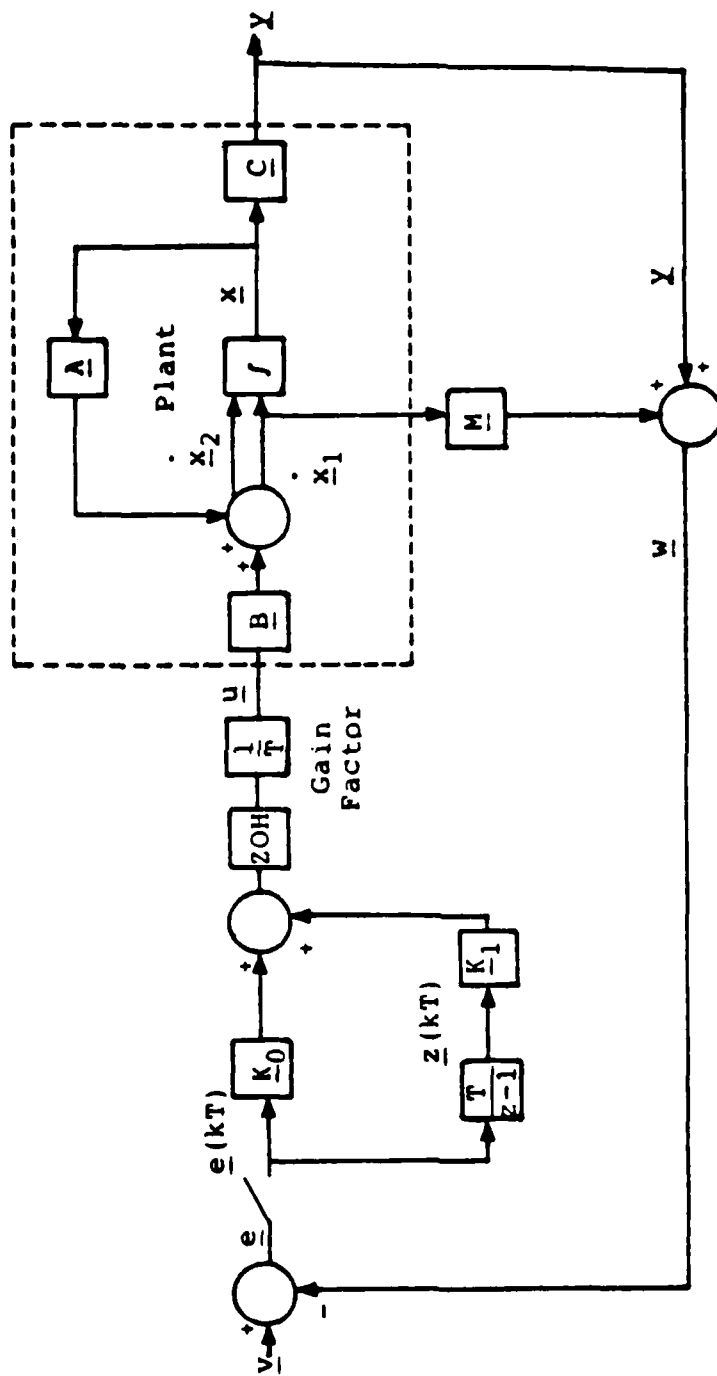


Fig. B.2. System Block Diagram--Discrete Case

where the forward path gain g equals the sampling frequency, $(1/T)$. The $\underline{z}(kT)$ matrix is derived from the backward difference equation,

$$\underline{z}[(k+1)T] = \underline{z}(kT) + T\underline{e}(kT) \quad (B-6)$$

The steps to be taken next in the design method depend on whether or not the first Markov parameter $[\underline{CB}]$, has full rank, i.e., does it have an inverse. If the matrix $[\underline{CB}]$ has full rank, the plant is called "regular" and no measurement matrix \underline{M} is needed. However, if $[\underline{CB}]$ does not have full rank, the plant is called "irregular" and \underline{M} is needed to form a new matrix $[\underline{FB}]$ (See Equations (B-12) through (B-14)) which does have an inverse. This is explained in more detail in the next sections. When the partitioned \underline{B} matrix in Equation (B-2a) has the form

$$\begin{bmatrix} 0 \\ \hline \underline{B}_2 \end{bmatrix} \quad (B-7)$$

then

$$[\underline{CB}] = [\underline{C}_2 \underline{B}_2] \quad (B-8)$$

$$[\underline{FB}] = [\underline{F}_2 \underline{B}_2] \quad (B-9)$$

As in the continuous case, a q -dimensional bank of integrators applies equally well to the discrete design (Figure B.2).

Asymptotic Characteristics

As the gain factor of the system, g (or $1/T$ for the discrete case), approaches infinity, the system transfer function matrix $G(s)$ assumes the asymptotic form

$$\underline{\Gamma}(\lambda) = \tilde{\underline{\Gamma}}(\lambda) + \hat{\underline{\Gamma}}(\lambda) \quad (\text{B-10})$$

where

$\tilde{\underline{\Gamma}}(\lambda)$ is the slow transfer function matrix

$\hat{\underline{\Gamma}}(\lambda)$ is the fast transfer function matrix

The roots of the asymptotic closed-loop transfer function may be grouped into three sets: \underline{Z}_1 , \underline{Z}_2 , and \underline{Z}_3 . Table B.1 gives the equations for finding these asymptotic roots. Sets \underline{Z}_1 and \underline{Z}_2 correspond to the slow modes of the system, where the modes associated with the roots in \underline{Z}_1 become uncontrollable, and, for regular designs, the modes associated with the roots in \underline{Z}_2 become unobservable as the gain increases. Set \underline{Z}_3 , the infinite roots, are associated with the fast modes of the system which become dominant as the gain increases.

The roots in set \underline{Z}_2 correspond to the transmission zeros of the system which are not altered by output feedback. As the gain is increased, the closed-loop roots of the system tend to migrate toward the transmission zeros. This may adversely affect the system stability if the location of these zeros is in the unstable region.

TABLE B.1

ASYMPTOTIC EQUATIONS FOR ZERO- \underline{B}_2 FORM

System represented by:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ \underline{B}_2 \end{bmatrix} u \quad \text{and} \quad y = \begin{bmatrix} \hat{C}_1 & \hat{C}_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Continuous Case
(s-plane)

Gain Factor = g

$$\hat{P}(s) = \underline{B}_0 (\lambda \underline{E}_n - \hat{A}_0)^{-1} \underline{B}_0$$

$$\hat{P}(s) = (\lambda \underline{E}_n + g \hat{C}_2 \hat{B}_2 \hat{K}_0)^{-1} g \hat{C}_1 \hat{B}_2 \hat{K}_0$$

Finite Roots

$$\underline{E}_1 = (\lambda \underline{E}_0 - \underline{K}_1 = 0)$$

$$\underline{E}_2 = (\lambda \underline{E}_{n-m} - \hat{A}_{11} - \hat{A}_{12} \hat{E}_2^{-1} \hat{E}_1 = 0)$$

Infinite Roots

$$\underline{E}_3 = (\lambda \underline{E}_n + g \hat{C}_2 \hat{B}_2 \hat{K}_0 = 0)$$

Discrete Case
(z-plane)

Gain Factor = 1/T

$$\hat{P}(z) = \underline{B}_0 (\lambda \underline{E}_n - \hat{A}_0)^{-1} \underline{B}_0$$

$$\hat{P}(z) = (\lambda \underline{E}_n - \underline{E}_m + \hat{C}_2 \hat{B}_2 \hat{K}_0)^{-1} \hat{C}_1 \hat{B}_2 \hat{K}_0$$

$$\underline{E}_1 = (\lambda \underline{E}_m - \underline{E}_m + \hat{C}_2 \hat{B}_2 \hat{K}_0 = 0)$$

$$\underline{E}_2 = (\lambda \underline{E}_{n-m} - \hat{A}_{11} - \hat{A}_{12} \hat{E}_2^{-1} \hat{E}_1 = 0)$$

$$\underline{E}_3 = (\lambda \underline{E}_n - \underline{E}_m - \hat{C}_2 \hat{B}_2 \hat{K}_0 = 0)$$

where

$$\hat{A}_0 = \begin{bmatrix} -\hat{K}_0^{-1} \hat{K}_1 & 1 \\ \hat{A}_{12} \hat{E}_2^{-1} \hat{K}_0^{-1} \hat{K}_1 & \hat{A}_{11} - \hat{A}_{12} \hat{E}_2^{-1} \hat{E}_1 \end{bmatrix}$$

$$\hat{B}_0 = \begin{bmatrix} 1 \\ \hat{A}_{12} \hat{E}_2^{-1} \end{bmatrix}$$

Regular Design

$$\underline{E}_0 = \begin{bmatrix} \hat{K}_0^{-1} \hat{K}_1 & 1 \end{bmatrix}$$

Irregular Design

$$\underline{E}_0 = \begin{bmatrix} \hat{A}_{12} \hat{E}_2^{-1} \hat{K}_0^{-1} \hat{K}_1 & \hat{A}_{11} - \hat{A}_{12} \hat{E}_2^{-1} \hat{E}_1 \end{bmatrix}$$

Reference 20 gives a procedure for locating the transmission zeros of a system.

For a regular design, as the gain increases, the system output responses become increasingly decoupled and dominated by the infinite root characteristics. The asymptotic closed-loop transfer function for the continuous case has the form

$$\underline{\Gamma}(\lambda) = \text{diag} \left\{ \frac{g\sigma_1}{\lambda + g\sigma_1}, \frac{g\sigma_2}{\lambda + g\sigma_2}, \dots, \frac{g\sigma_\ell}{\lambda + g\sigma_\ell} \right\} \quad (\text{B-11})$$

For the discrete case the form is

$$\underline{\Gamma}(\lambda) = \text{diag} \left\{ \frac{\sigma_1}{\lambda - 1 + \sigma_1}, \frac{\sigma_2}{\lambda - 1 + \sigma_2}, \dots, \frac{\sigma_\ell}{\lambda - 1 + \sigma_\ell} \right\} \quad (\text{B-12})$$

where the σ_i ($i = 1, \dots, \ell$) are determined by the weighting matrix, \underline{W} .

For certain irregular designs where the structure of the output vector creates a diagonal Γ matrix, the system will exhibit increasingly decoupled behavior (Chapter IV). In other cases, the Γ matrix contains off-diagonal terms which prevent full output decoupling as the gain approaches infinity. In all irregular designs, the transmission zero always appears as a finite asymptotic root in at least one position on the diagonal of Γ and may appear on the off-diagonal. This characteristic places an upper bound on the time responses of these particular outputs (18).

Regular Plant

For the system to be classified as "regular" the first Markov parameter $[CB]$ must have full rank. If this is true, the gain matrices can be found from

$$\underline{K}_0 = [CB]^{-1} \underline{\Sigma} \quad (B-13)$$

and

$$\underline{K}_1 = \bar{\alpha} [CB]^{-1} \underline{\Sigma} \quad (B-14)$$

where

$\bar{\alpha}$ is a constant which assigns the ratio of proportional to integral control

$\underline{\Sigma}$ is the diagonal weighting matrix

The diagonal weighting matrix, $\underline{\Sigma} = \text{diag} \{ \sigma_1, \sigma_2, \dots, \sigma_\ell \}$, is specified by the designer. Each σ_i ($i=1, \dots, \ell$) determines the weighting of the effect of a particular error signal on each control input. This is the methodology used in the MULTI design program and is a simplified version of the complete Porter method. In theory, the total number of finite (slow) roots of the system is equal to:

$$Z_f = n + q\ell - \ell \quad (B-15)$$

which also equals $\underline{Z}_1 + \underline{Z}_2$ (Table B.1).

The \underline{Z}_1 roots, equal to $(q\ell)$ in number, are assigned by the relationship between the proportional and integral matrices. If the matrices differ by a simple

Proportionality constant, $\bar{\alpha}$, then all of the \underline{z}_1 roots are assigned, under conditions of infinite gain (asymptotically), to the value of $-\bar{\alpha}$ in the s-plane. By replacing $\bar{\alpha}$ with a diagonal matrix, these roots can be individually assigned as the negative value of its diagonal elements.

Irregular Plant

If the first Markov parameter $[\underline{CB}]$ is rank deficient, then the plant is called "irregular." In this case, the \underline{C} matrix must be replaced by

$$\underline{F} = [\underline{F}_1 \mid \underline{F}_2] \quad (\text{B-16})$$

where

$$\underline{F}_1 = [\underline{C}_1 + \underline{MA}_{11}] \quad (\text{B-17})$$

$$\underline{F}_2 = [\underline{C}_2 + \underline{MA}_{12}] \quad (\text{B-18})$$

The matrix \underline{M} in the above equations is a measurement matrix which is chosen such that the matrix $[\underline{FB}]$ has full rank. The designer chooses the measurement matrix so that it is as sparse as possible, thus the smallest number of additional measurements are required. Reference 18 gives an approach for selecting the measurement matrix to achieve optimal decoupling. Once \underline{M} is formed, \underline{K}_0 and \underline{K}_1 are computed by

$$\underline{K}_0 = [\underline{FB}]^{-1} \underline{I} \quad (\text{B-19})$$

$$\underline{K}_1 = \bar{\alpha}[\underline{FB}]^{-1}\underline{Z} \quad (\text{B-20})$$

which are similar to Equations (B-13) and (B-14). As in the regular design case, the same conditions of \underline{Z}_1 root assignment apply here.

For irregular plants the error vector \underline{e} is defined as

$$\underline{e} = \underline{v} - \underline{w} \quad (\text{B-21})$$

where

$$\underline{w} = \underline{y} + \underline{M}\dot{\underline{x}}_1 \quad (\text{B-22})$$

For step inputs the values of the rates, $\dot{\underline{x}}_1$, become zero in the steady state because they represent kinematic variables (no \underline{B} matrix entries).

The computer program MULTI greatly reduces the time required to achieve a satisfactory design. The MULTI User's Manual (13) describes the program and its operation.

Appendix C: Aero Data and State Space Matrices

Aeronautical data in nondimensional body axis form are presented for both the longitudinal and lateral modes of the F-15/STOL at each of the four flight conditions evaluated in this thesis. In addition, the state space forms for the four-state model are also included. These matrices are given in dimensionalized, 1/rad, body axis form except for the B matrix which is in units of 1/deg.

The three-state models used for the constant g pull-up maneuver are formed from the four-state model by setting $u = 0$ (short-period approximation). The three-state models are presented with the same units as described above.

TABLE C.1
LONGITUDINAL STATE SPACE MATRICES (FOUR-STATE MODEL)

Flight Condition: 0.3 Mach/FL 200

A Matrix

.0000E+00	.0000E+00	.1000E+01	.0000E+00
-.3147E+02	-.3954E-01	-.6599E+02	-.3589E+01
.0000E+00	-.1157E-03	-.7350E+00	.7840E+00
-.2194E-01	-.8427E-05	.9772E+00	-.4647E+00

B Matrix

.0000E+00	.0000E+00	.0000E+00
-.2642E-01	-.4657E-01	.2000E+02
.2084E-01	-.3677E-01	.0000E+00
-.3154E-03	-.1119E-02	.0000E+00

C Matrix

.1000E+01	.0000E+00	.0000E+00	.0000E+00
.0000E+00	.1000E+01	.0000E+00	.0000E+00
.1000E+01	.0000E+00	.0000E+00	-.1000E+01

Notes: (All variables are perturbation quantities and listed in order.)

1. States are pitch angle, velocity, pitch rate, and angle of attack.
2. Control inputs are canard, stabilator, and throttle.
3. Outputs are pitch angle, velocity, and flight path.

TABLE C.2

LONGITUDINAL STATE SPACE MATRICES (FOUR-STATE MODEL)

Flight Condition: 0.9 Mach/FL 200

A Matrix

.0000E+00	.0000E+00	.1000E+01	.0000E+00
-.3219E+02	-.1789E-01	-.2002E+02	.2790E+02
.0000E+00	-.2357E-03	-.2002E+01	.1082E+02
-.7398E-03	-.3476E-06	.9998E+00	-.1497E+01

B Matrix

.0000E+00	.0000E+00	.0000E+00
.2494E-01	-.1741E+00	.4500E+02
.2047E+00	-.3440E+00	.0000E+00
-.7908E-03	-.3278E-02	.0000E+00

C Matrix

.1000E+01	.0000E+00	.0000E+00	.0000E+00
.0000E+00	.1000E+01	.0000E+00	.0000E+00
.1000E+01	.0000E+00	.0000E+00	-.1000E+01

Notes: (All variables are perturbation quantities and listed in order.)

1. States are pitch angle, velocity, pitch rate, and angle of attack.
2. Control inputs are canard, stabilator, and throttle.
3. Outputs are pitch angle, velocity, and flight path.

TABLE C.3
LONGITUDINAL STATE SPACE MATRICES (THREE-STATE MODEL)

Flight Condition: 0.9 Mach/FL 200

A Matrix

.0000E+00	.1000E+01	.0000E+00
.0000E+00	-.2002E+01	.1082E+02
-.7398E-03	.9998E+00	-.1497E+01

B Matrix

.0000E+00		.0000E+00
.2047E+00		-.3838E+00
-.7908E-03		-.3544E-02

C Matrix

.1000E+01	.0000E+00	.0000E+00
.0000E+00	.0000E+00	.1000E+01

Notes: (All variables are perturbation quantities and listed in order.)

1. States are pitch angle, pitch rate, and angle of attack.
2. Control inputs are canard and stabilator-nozzle.
3. Outputs are pitch angle and angle of attack.

TABLE C.4

LONGITUDINAL STATE SPACE MATRICES (FOUR-STATE MODEL)

Flight Condition: 1.4 Mach/FL 200

A Matrix

.0000E+00	.0000E+00	.1000E+01	.0000E+00
-.3220E+02	-.1747E-01	.4613E+01	.4292E+02
.0000E+00	.1283E-02	-.5612E+01	-.5384E+02
.7043E-04	-.9365E-05	.1000E+01	-.1787E+01

B Matrix

.0000E+00	.0000E+00	.0000E+00
.1854E-01	-.3944E+00	.1200E+03
.2318E+00	-.5847E+00	.0000E+00
-.3812E-03	-.3300E-02	.0000E+00

C Matrix

.1000E+01	.0000E+00	.0000E+00	.0000E+00
.0000E+00	.1000E+01	.0000E+00	.0000E+00
.1000E+01	.0000E+00	.0000E+00	-.1000E+01

Notes: (All variables are perturbation quantities and listed in order.)

1. States are pitch angle, velocity, pitch rate, and angle of attack.
2. Control inputs are canard, stabilator, and throttle.
3. Outputs are pitch angle, velocity, and flight path.

TABLE C.5
LONGITUDINAL STATE SPACE MATRICES (THREE-STATE MODEL)

Flight Condition: 1.4 Mach/FL 200

A Matrix

.0000E+00	.1000E+01	.0000E+00
.0000E+00	-.5612E+01	-.5384E+02
-.7044E-04	.1000E+01	-.1787E+01

B Matrix

.0000E+00		.0000E+00
.2318E+00		-.7587E+00
-.3812E-03		-.3867E-02

C Matrix

.1000E+01	.0000E+00	.0000E+00
.0000E+00	.0000E+00	.1000E+01

Notes: (All variables are perturbation quantities and listed in order.)

1. States are pitch angle, pitch rate, and angle of attack.
2. Control inputs are canard and stabilator-nozzle.
3. Outputs are pitch angle and angle of attack.

TABLE C.6
LONGITUDINAL STATE SPACE MATRICES (FOUR-STATE MODEL)

Flight Condition: 2.0 Mach/FL 400			
<u>A Matrix</u>			
.0000E+00	.0000E+00	.1000E+01	.0000E+00
-.3220E+02	-.1477E-01	-.5135E+01	-.3192E+01
.0000E+00	-.1954E-02	-.8888E+01	-.2469E+02
-.4384E-04	.6193E-05	.1000E+01	-.8424E+00
<u>B Matrix</u>			
.0000E+00	.0000E+00	.0000E+00	.0000E+00
-.3349E-01	-.2658E+00	.1000E+03	
.1237E+00	-.3690E+00	.0000E+00	
-.3684E-03	-.1193E-02	.0000E+00	
<u>C Matrix</u>			
.1000E+01	.0000E+00	.0000E+00	.0000E+00
.0000E+00	.1000E+01	.0000E+00	.0000E+00
.1000E+01	.0000E+00	.0000E+00	-.1000E+01

Notes: (All variables are perturbation quantities and listed in order.)

1. States are pitch angle, velocity, pitch rate, and angle of attack.
2. Control inputs are canard, stabilator, and throttle.
3. Outputs are pitch angle, velocity, and flight path.

TABLE C.7
LONGITUDINAL STATE SPACE MATRICES (THREE-STATE MODEL)

Flight Condition: 2.0 Mach/FL 400

A Matrix

.0000E+00	.1000E+01	.0000E+00
.0000E+00	-.8888E+01	-.2469E+02
-.4384E-04	.1000E+01	-.8424E+00

B Matrix

.0000E+00		.0000E+00
.1237E+00		-.5562E+00
-.3684E-03		-.1618E-02

C Matrix

.1000E+01	.0000E+00	.0000E+00
.0000E+00	.0000E+00	.1000E+01

Notes: (All variables are perturbation quantities and listed in order.)

1. States are pitch angle, pitch rate, and angle of attack.
2. Control inputs are canard and stabilator-nozzle.
3. Outputs are pitch angle and angle of attack.

Aeronautical Data

0.3 Mach/FL 200

AIRCRAFT PARAMETERS

Q (DYNAMIC PRESSURE - LBS/FT**2) = 61.3429
 S (WING REFERENCE AREA - FT**2) = 608.000
 C (WING MEAN AERODYNAMIC CORD - FT) = 15.9400
 B (WING SPAN - FT) = 42.7000
 VT (TRIM VELOCITY - FT/SEC) = 311.178
 THETA = 12.2435
 W (WEIGHT - LBS) = 37794.2
 IXX (SLUG-FT**2) = 25938.0
 IYY (SLUG-FT**2) = 185287.
 IZZ (SLUG-FT**2) = 206359.
 IXZ (SLUG-FT**2) = -2543.00

ALPHA = 12.2435

LONGITUDINAL NON-DIM BODY AXIS COEFFICIENTS (1/DEG)

CZA = -.794180E-01	CZD3 = -.342080E-02
CZQ = 0.	CZD4 = -.342080E-02
CZH = .636480E-04	CZD5 = 0.
CZU = -.128400E-01	CZD6 = 0.
CZD1 = -.308820E-02	CZD7 = 0.
CZD2 = -.109620E-01	CZD8 = 0.
CMA = .426440E-02	CMD3 = -.365180E-02
CMQ = -.156100	CMD4 = -.365180E-02
CMH = -.278180E-04	CMD5 = 0.
CMU = -.561190E-02	CMD6 = 0.
CMD1 = .649500E-02	CMD7 = 0.
CMD2 = -.114630E-01	CMD8 = 0.
CXA = -.197150E-02	CXD3 = .592340E-04
CXQ = 0.	CXD4 = .592340E-04
CXH = .959480E-03	CXD5 = 0.
CXU = -.193600	CXD6 = 0.
CXD1 = -.831580E-03	CXD7 = 0.
CXD2 = -.146560E-02	CXD8 = 0.

LONGITUDINAL AXIS DIMENSIONAL DERIVATIVES BODY AXIS (1/RAD)

ZA = -144.591	ZD3 = -6.22801
ZQ = 0.	ZD4 = -6.22801
ZH = .649943E-05	ZD5 = 0.
ZU = -.262232E-02	ZD6 = 0.
ZD1 = -5.62247	ZD7 = 0.
ZD2 = -19.9578	ZD8 = 0.

MA =	.783956	MD3 =	-.671337
MQ =	-.734998	MD4 =	-.671337
MH =	-.286832E-06	MD5 =	0.
MU =	-.115729E-03	MD6 =	0.
MD1 =	1.19402	MD7 =	0.
MD2 =	-2.10733	MD8 =	0.

XA =	-3.58937	XD3 =	.107843
XQ =	0.	XD4 =	.107843
XH =	.979775E-04	XD5 =	0.
XU =	-.395390E-01	XD6 =	0.
XD1 =	-1.51400	XD7 =	0.
XD2 =	-2.66832	XD8 =	0.

LAT-DIR NON-DIM BODY AXIS COEFFICIENTS (1/DEG)

CLB =	-.254330E-02	CLD4 =	.683140E-03
CLP =	-.534780E-02	CLD5 =	.711050E-03
CLR =	.382830E-02	CLD6 =	.184410E-03
CLD1 =	.742100E-04	CLD7 =	-.184410E-03
CLD2 =	-.184590E-04	CLD8 =	0.
CLD3 =	.885870E-03	CLD9 =	0.

CNB =	-.483370E-03	CND4 =	.638610E-04
CNP =	-.232640E-02	CND5 =	0.
CNR =	-.893800E-02	CND6 =	0.
CND1 =	-.144010E-02	CND7 =	0.
CND2 =	.565260E-03	CND8 =	0.
CND3 =	.368420E-03	CND9 =	0.

CYB =	-.167190E-01	CYD4 =	-.845490E-04
CYP =	0.	CYD5 =	0.
CYR =	0.	CYD6 =	0.
CYD1 =	.317720E-02	CYD7 =	0.
CYD2 =	.131870E-02	CYD8 =	0.
CYD3 =	-.101060E-02	CYD9 =	0.

LAT-DIR BODY AXIS DIMENSIONAL DERIVATIVES (1/RAD)

NB =	-.213735	ND4 =	.282378E-01
NP =	-.705778E-01	ND5 =	0.
NR =	-.271159	ND6 =	0.
ND1 =	-.636777	ND7 =	0.
ND2 =	.249944	ND8 =	0.
ND3 =	.162906	ND9 =	0.

LB =	-8.94704	LD4 =	2.40321
LP =	-1.29076	LD5 =	2.50139
LR =	.924010	LD6 =	.648734
LD1 =	.261062	LD7 =	-.648734
LD2 =	-.649367E-01	LD8 =	0.
LD3 =	3.11639	LD9 =	0.

YB =	-30.4391	YD4 =	-.153932
YP =	0.	YD5 =	0.
YR =	0.	YD6 =	0.
YD1 =	5.78451	YD7 =	0.
YD2 =	2.40087	YD8 =	0.
YD3 =	-1.83993	YD9 =	0.

WHERE:

LONGITUDINAL MODE

D1 = CANARD	D5 = ROTATING VANE, RT TOP
D2 = STABILATOR	D6 = ROTATING VANE, RT BOT
D3 = LEFT 2-D NOZZLE	D7 = ROTATING VANE, LT TOP
D4 = RIGHT 2-D NOZZLE	D8 = ROTATING VANE, LT BOT

LATERAL MODE

D1 = RUDDER	D5 = DIFFERENTIAL FLAPS
D2 = DIFFERENTIAL CANARD	D6 = LEFT 2-D NOZZLE
D3 = DIFFERENTIAL STABILATOR	D7 = RIGHT 2-D NOZZLE
D4 = DIFFERENTIAL AILERON	

Aeronautical Data

0.9 Mach/FL 200

AIRCRAFT PARAMETERS

Q (DYNAMIC PRESSURE - LBS/FT**2) = 552.086
 S (WING REFERENCE AREA - FT**2) = 608.000
 C (WING MEAN AERODYNAMIC CORD - FT) = 15.9400
 B (WING SPAN - FT) = 42.7000
 VT (TRIM VELOCITY - FT/SEC) = 933.534
 THETA = 1.22900
 W (WEIGHT - LBS) = 37794.2
 IXX (SLUG-FT**2) = 25938.0
 IYY (SLUG-FT**2) = 185287.
 IZZ (SLUG-FT**2) = 206359.
 IXZ (SLUG-FT**2) = -2543.00

ALPHA = 1.22900

LONGITUDINAL NON-DIM BODY AXIS COEFFICIENTS (1/DEG)

CZA = -.852800E-01	CZD3 = -.643690E-03
CZQ = 0.	CZD4 = -.643690E-03
CZH = .787590E-05	CZD5 = 0.
CZU = -.529610E-03	CZD6 = 0.
CZD1 = -.258140E-02	CZD7 = 0.
CZD2 = -.106980E-01	CZD8 = 0.
CMA = .654210E-02	CMD3 = -.102220E-02
CMQ = -.141700	CMD4 = -.102220E-02
CMH = -.566610E-04	CMD5 = 0.
CMU = -.381010E-02	CMD6 = 0.
CMD1 = .709050E-02	CMD7 = 0.
CMD2 = -.119130E-01	CMD8 = 0.
CXA = .170280E-02	CXD3 = -.103160E-04
CXQ = 0.	CXD4 = -.103160E-04
CXH = .434250E-03	CXD5 = 0.
CXU = -.292010E-01	CXD6 = 0.
CXD1 = .872020E-04	CXD7 = 0.
CXD2 = -.608630E-03	CXD8 = 0.

LONGITUDINAL AXIS DIMENSIONAL DERIVATIVES BODY AXIS (1/RAD)

ZA = -1397.37	ZD3 = -10.5473
ZQ = 0.	ZD4 = -10.5473
ZH = .241275E-05	ZD5 = 0.
ZU = -.324487E-03	ZD6 = 0.
ZD1 = -42.2980	ZD7 = 0.
ZD2 = -175.294	ZD8 = 0.

MA =	10.8241	MD3 =	-1.69127
MQ =	-2.00159	MD4 =	-1.69127
MH =	-.175270E-03	MD5 =	0.
MU =	-.235716E-03	MD6 =	0.
MD1 =	11.7315	MD7 =	0.
MD2 =	-19.7105	MD8 =	0.
XA =	27.9015	XD3 =	-.169035
XQ =	0.	XD4 =	-.169035
XH =	.133031E-03	XD5 =	0.
XU =	-.178912E-01	XD6 =	0.
XD1 =	1.42886	XD7 =	0.
XD2 =	-9.97281	XD8 =	0.

LAT-DIR NON-DIM BODY AXIS COEFFICIENTS (1/DEG)

CLB =	-.122900E-02	CLD4 =	.342770E-03
CLP =	-.512830E-02	CLD5 =	.662010E-03
CLR =	.107820E-02	CLD6 =	.910410E-04
CLD1 =	.381080E-04	CLD7 =	-.910410E-04
CLD2 =	-.106700E-03	CLD8 =	0.
CLD3 =	.788040E-03	CLD9 =	0.
CNB =	.237600E-02	CND4 =	.672630E-04
CNP =	-.203200E-03	CND5 =	0.
CNR =	-.865630E-02	CND6 =	0.
CND1 =	-.124030E-02	CND7 =	0.
CND2 =	.435700E-03	CND8 =	0.
CND3 =	.502250E-03	CND9 =	0.
CYB =	-.210860E-01	CYD4 =	-.145770E-03
CYP =	0.	CYD5 =	0.
CYR =	0.	CYD6 =	0.
CYD1 =	.284130E-02	CYD7 =	0.
CYD2 =	.552720E-03	CYD8 =	0.
CYD3 =	-.127680E-02	CYD9 =	0.

LAT-DIR BODY AXIS DIMENSIONAL DERIVATIVES (1/RAD)

NB =	9.45549	ND4 =	.267679
NP =	-.184939E-01	ND5 =	0.
NR =	-.787839	ND6 =	0.
ND1 =	-4.93587	ND7 =	0.
ND2 =	1.73390	ND8 =	0.
ND3 =	1.99874	ND9 =	0.
LB =	-38.9113	LD4 =	10.8524
LP =	-3.71335	LD5 =	20.9599
LR =	.780713	LD6 =	2.88245
LD1 =	1.20654	LD7 =	-2.88245
LD2 =	-3.37823	LD8 =	0.
LD3 =	24.9501	LD9 =	0.

YB =	-345.508	YD4 =	-2.38854
YP =	0.	YD5 =	0.
YR =	0.	YD6 =	0.
YD1 =	46.5566	YD7 =	0.
YD2 =	9.05669	YD8 =	0.
YD3 =	-20.9212	YD9 =	0.

.....

WHERE:

LONGITUDINAL MODE

D1 = CANARD	D5 = ROTATING VANE, RT TOP
D2 = STABILATOR	D6 = ROTATING VANE, RT BOT
D3 = LEFT 2-D NOZZLE	D7 = ROTATING VANE, LT TOP
D4 = RIGHT 2-D NOZZLE	D8 = ROTATING VANE, LT BOT

LATERAL MODE

D1 = RUDDER	D5 = DIFFERENTIAL FLAPS
D2 = DIFFERENTIAL CANARD	D6 = LEFT 2-D NOZZLE
D3 = DIFFERENTIAL STABILATOR	D7 = RIGHT 2-D NOZZLE
D4 = DIFFERENTIAL AILERON	

Aeronautical Data

1.4 Mach/FL 200

AIRCRAFT PARAMETERS

Q (DYNAMIC PRESSURE - LBS/FT**2) = 1335.91
 S (WING REFERENCE AREA - FT**2) = 608.000
 C (WING MEAN AERODYNAMIC CORD - FT) = 15.9400
 B (WING SPAN - FT) = 42.7000
 VT (TRIM VELOCITY - FT/SEC) = 1452.16
 THETA = -.182000
 W (WEIGHT - LBS) = 37794.2
 IXX (SLUG-FT**2) = 25938.0
 IYY (SLUG-FT**2) = 185287.
 IZZ (SLUG-FT**2) = 206359.
 IXZ (SLUG-FT**2) = -2543.00

ALPHA = -.182000

LONGITUDINAL NON-DIM BODY AXIS COEFFICIENTS (1/DEG)

CZA = -.654580E-01	CZD3 = -.889950E-03
CZQ = 0.	CZD4 = -.889950E-03
CZH = .330080E-03	CZD5 = 0.
CZU = -.142690E-01	CZD6 = 0.
CZD1 = -.799900E-03	CZD7 = 0.
CZD2 = -.692730E-02	CZD8 = 0.
CMA = -.134490E-01	CMD3 = -.186670E-02
CMQ = -.255400	CMD4 = -.186670E-02
CMH = .308320E-03	CMD5 = 0.
CMU = .133280E-01	CMD6 = 0.
CMD1 = .331640E-02	CMD7 = 0.
CMD2 = -.836780E-02	CMD8 = 0.
CXA = .108250E-02	CXD3 = -.337950E-04
CXQ = 0.	CXD4 = -.337950E-04
CXH = .424060E-03	CXD5 = 0.
CXU = -.183310E-01	CXD6 = 0.
CXD1 = .267760E-04	CXD7 = 0.
CXD2 = -.536760E-03	CXD8 = 0.

LONGITUDINAL AXIS DIMENSIONAL DERIVATIVES BODY AXIS (1/RAD)

ZA = -2595.36	ZD3 = -35.2859
ZQ = 0.	ZD4 = -35.2859
ZH = .157295E-03	ZD5 = 0.
ZU = -.135994E-01	ZD6 = 0.
ZD1 = -31.7153	ZD7 = 0.
ZD2 = -274.662	ZD8 = 0.

MA	=	-53.8440	MD3	=	-7.47346
MQ	=	-5.61191	MD4	=	-7.47346
MH	=	.148358E-04	MD5	=	0.
MU	=	.128264E-02	MD6	=	0.
MD1	=	13.2774	MD7	=	0.
MD2	=	-33.5011	MD8	=	0.

XA	=	42.9204	XD3	=	-1.33995
XQ	=	0.	XD4	=	-1.33995
XH	=	.202080E-03	XD5	=	0.
XU	=	-.174708E-01	XD6	=	0.
XD1	=	1.06165	XD7	=	0.
XD2	=	-21.2821	XD8	=	0.

LAT-DIR NON-DIM BODY AXIS COEFFICIENTS (1/DEG)

CLB	=	-.647440E-03	CLD4	=	.456370E-04
CLP	=	-.551230E-02	CLD5	=	.675220E-03
CLR	=	.910580E-03	CLD6	=	.346170E-04
CLD1	=	.292080E-04	CLD7	=	-.346170E-04
CLD2	=	-.562730E-03	CLD8	=	0.
CLD3	=	.666430E-03	CLD9	=	0.

CNB	=	.946930E-03	CND4	=	.429460E-05
CNP	=	.156840E-04	CND5	=	0.
CNR	=	-.153450E-01	CND6	=	-.695530E-06
CND1	=	-.364710E-03	CND7	=	.695530E-06
CND2	=	.436740E-04	CND8	=	0.
CND3	=	.537130E-04	CND9	=	0.

CYB	=	-.155410E-01	CYD4	=	-.181700E-04
CYP	=	0.	CYD5	=	0.
CYR	=	0.	CYD6	=	0.
CYD1	=	.747910E-03	CYD7	=	0.
CYD2	=	.365130E-03	CYD8	=	0.
CYD3	=	-.452300E-03	CYD9	=	0.

LAT-DIR BODY AXIS DIMENSIONAL DERIVATIVES (1/RAD)

NB	=	9.11856	ND4	=	.413553E-01
NP	=	.222048E-02	ND5	=	0.
NR	=	-2.17249	ND6	=	-.669768E-02
ND1	=	-3.51201	ND7	=	.669768E-02
ND2	=	.420563	ND8	=	0.
ND3	=	.517235	ND9	=	0.

LB	=	-49.6015	LD4	=	3.49633
LP	=	-6.20884	LD5	=	51.7298
LR	=	1.02564	LD6	=	2.65207
LD1	=	2.23768	LD7	=	-2.65207
LD2	=	-43.1117	LD8	=	0.
LD3	=	51.0564	LD9	=	0.

YB =	-616.190	YD4 =	-.720427
YP =	0.	YD5 =	0.
YR =	0.	YD6 =	0.
YD1 =	29.6541	YD7 =	0.
YD2 =	14.4771	YD8 =	0.
YD3 =	-17.9334	YD9 =	0.

WHERE:

LONGITUDINAL MODE

D1 = CANARD	D5 = ROTATING VANE, RT TOP
D2 = STABILATOR	D6 = ROTATING VANE, RT BOT
D3 = LEFT 2-D NOZZLE	D7 = ROTATING VANE, LT TOP
D4 = RIGHT 2-D NOZZLE	D8 = ROTATING VANE, LT BOT

LATERAL MODE

D1 = RUDDER	D5 = DIFFERENTIAL FLAPS
D2 = DIFFERENTIAL CANARD	D6 = LEFT 2-D NOZZLE
D3 = DIFFERENTIAL STABILATOR	D7 = RIGHT 2-D NOZZLE
D4 = DIFFERENTIAL AILERON	

Aeronautical Data

2.0 Mach/FL 400

AIRCRAFT PARAMETERS

Q (DYNAMIC PRESSURE - LBS/FT**2) = 1104.44
 S (WING REFERENCE AREA - FT**2) = 608.000
 C (WING MEAN AERODYNAMIC CORD - FT) = 15.9400
 B (WING SPAN - FT) = 42.7000
 VT (TRIM VELOCITY - FT/SEC) = 1942.00
 THETA = .151500
 W (WEIGHT - LBS) = 37794.2
 IXX (SLUG-FT**2) = 25938.0
 IYY (SLUG-FT**2) = 185287.
 IZZ (SLUG-FT**2) = 206359.
 IXZ (SLUG-FT**2) = -2543.00

ALPHA = .151500

LONGITUDINAL NON-DIM BODY AXIS COEFFICIENTS (1/DEG)

CZA = -.499060E-01	CZD3 = -.113090E-02
CZQ = 0.	CZD4 = -.113090E-02
CZH = -.735710E-03	CZD5 = 0.
CZU = .204110E-01	CZD6 = 0.
CZD1 = -.125090E-02	CZD7 = 0.
CZD2 = -.404930E-02	CZD8 = 0.
CMA = -.745840E-02	CMD3 = -.254300E-02
CMQ = -.654300	CMD4 = -.254300E-02
CMH = -.118360E-02	CMD5 = 0.
CMU = -.328360E-01	CMD6 = 0.
CMD1 = .214210E-02	CMD7 = 0.
CMD2 = -.638570E-02	CMD8 = 0.
CXA = -.973700E-04	CXD3 = -.231340E-05
CXQ = 0.	CXD4 = -.231340E-05
CXH = .903860E-03	CXD5 = 0.
CXU = -.250760E-01	CXD6 = 0.
CXD1 = -.585440E-04	CXD7 = 0.
CXD2 = -.464550E-03	CXD8 = 0.

LONGITUDINAL AXIS DIMENSIONAL DERIVATIVES BODY AXIS (1/RAD)

ZA = -1635.89	ZD3 = -37.0702
ZQ = 0.	ZD4 = -37.0702
ZH = -.216738E-03	ZD5 = 0.
ZU = .120261E-01	ZD6 = 0.
ZD1 = -41.0038	ZD7 = 0.
ZD2 = -132.734	ZD8 = 0.

MA = -24.6865	MD3 = -8.41704
MQ = -8.88790	MD4 = -8.41704
MH = -.352084E-04	MD5 = 0.
MU = -.195354E-02	MD6 = 0.
MD1 = 7.09011	MD7 = 0.
MD2 = -21.1360	MD8 = 0.
XA = -3.19173	XD3 = -.758319E-01
XQ = 0.	XD4 = -.758319E-01
XH = .266275E-03	XD5 = 0.
XU = -.147747E-01	XD6 = 0.
XD1 = -1.91904	XD7 = 0.
XD2 = -15.2277	XD8 = 0.

LAT-DIR NON-DIM BODY AXIS COEFFICIENTS (1/DEG)

CLB = -.760570E-03	CLD4 = .104500E-03
CLP = -.555170E-02	CLD5 = .671730E-03
CLR = .675400E-03	CLD6 = .376750E-04
CLD1 = .339610E-04	CLD7 = -.376750E-04
CLD2 = -.434310E-03	CLD8 = 0.
CLD3 = .460600E-03	CLD9 = 0.
CNB = .162300E-03	CND4 = .562860E-05
CNP = -.249140E-05	CND5 = 0.
CNR = -.375390E-01	CND6 = 0.
CND1 = -.275500E-03	CND7 = 0.
CND2 = .158240E-03	CND8 = 0.
CND3 = .247690E-04	CND9 = 0.
CYB = -.144930E-01	CYD4 = -.770730E-04
CYP = 0.	CYD5 = 0.
CYR = 0.	CYD6 = 0.
CYD1 = .498310E-03	CYD7 = 0.
CYD2 = .829290E-03	CYD8 = 0.
CYD3 = -.293180E-03	CYD9 = 0.

LAT-DIR BODY AXIS DIMENSIONAL DERIVATIVES (1/RAD)

NB = 1.29209	ND4 = .448100E-01
NP = -.218055E-03	ND5 = 0.
NR = -3.28553	ND6 = 0.
ND1 = -2.19329	ND7 = 0.
ND2 = 1.25977	ND8 = 0.
ND3 = .197189	ND9 = 0.
LB = -48.1726	LD4 = 6.61877
LP = -3.86577	LD5 = 42.5457
LR = .470296	LD6 = 2.38624
LD1 = 2.15101	LD7 = -2.38624
LD2 = -27.5081	LD8 = 0.
LD3 = 29.1733	LD9 = 0.

YB = -475.072	YD4 = -2.52641
YP = 0.	YD5 = 0.
YR = 0.	YD6 = 0.
YD1 = 16.3343	YD7 = 0.
YD2 = 27.1836	YD8 = 0.
YD3 = -9.61027	YD9 = 0.

WHERE:

LONGITUDINAL MODE

D1 = CANARD	D5 = ROTATING VANE, RT TOP
D2 = STABILATOR	D6 = ROTATING VANE, RT BOT
D3 = LEFT 2-D NOZZLE	D7 = ROTATING VANE, LT TOP
D4 = RIGHT 2-D NOZZLE	D8 = ROTATING VANE, LT BOT

LATERAL MODE

D1 = RUDDER	D5 = DIFFERENTIAL FLAPS
D2 = DIFFERENTIAL CANARD	D6 = LEFT 2-D NOZZLE
D3 = DIFFERENTIAL STABILATOR	D7 = RIGHT 2-D NOZZLE
D4 = DIFFERENTIAL AILERON	

Appendix D: Design Parameters and Response Plots

This appendix lists the remaining designs not detailed in Chapter V. Design parameters and response plots are given for the basic aircraft + actuators and the fully developed model (actuators + delay + sensors). The constant g pull-up (2.0 g) at 1.4 Mach/FL 200 is the only design that could not be achieved with all delays added. As a result, the final design includes only actuators and computational time delay.

The designs presented in this chapter include:

1. Direct Climb: 0.9 Mach/FL 200
1.4 Mach/FL 200
2.0 Mach/FL 400
2. Vertical Translation: 0.3 Mach/FL 200
1.4 Mach/FL 200
2.0 Mach/FL 400
3. Pitch Pointing: 0.3 Mach/FL 200
0.9 Mach/FL 200
2.0 Mach/FL 400
4. Constant g Pull-Up: 0.9 Mach/FL 200
1.4 Mach/FL 200

TABLE D.1

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Direct Climb (+2.0 degs)

Flight Condition: 0.9 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 0.8, 0.03491, 20, 20$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0.8, 0.03491, 20, 20$

Plant + Actuators

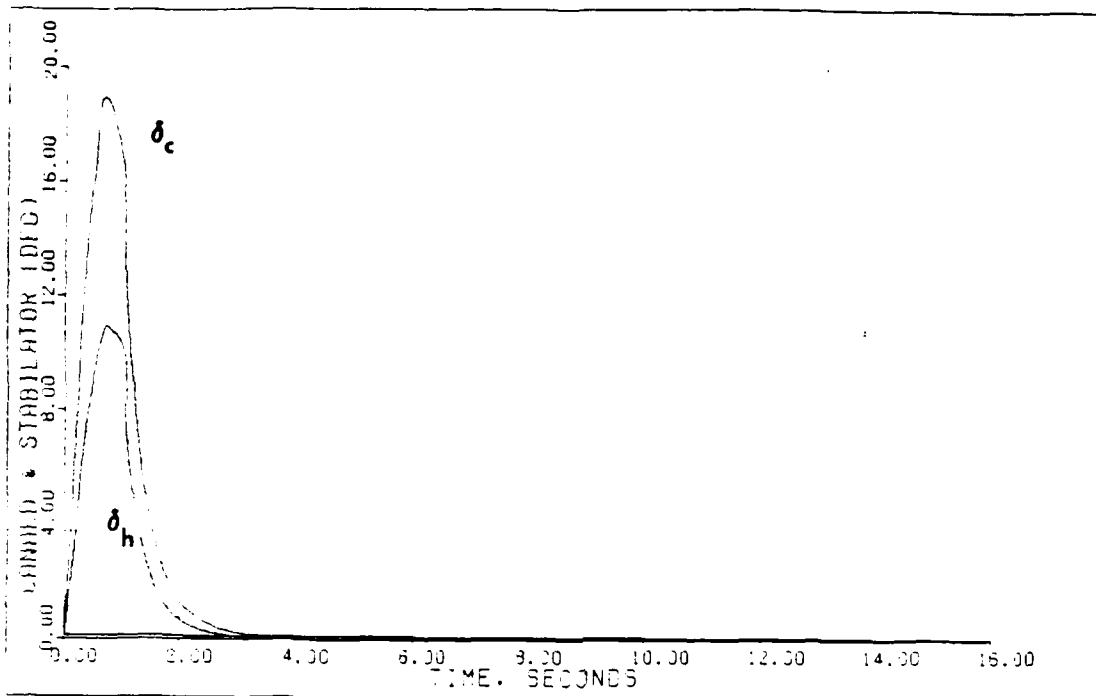
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
1.250	0.960	1.0	.1112E+02	.0000E+00	.2801E+02
		0.1	-.2683E+01	.0000E+00	.1667E+02
		0.08	-.1655E-01	.2133E-02	.4897E-01

Plant + Actuators + Delay + Sensors

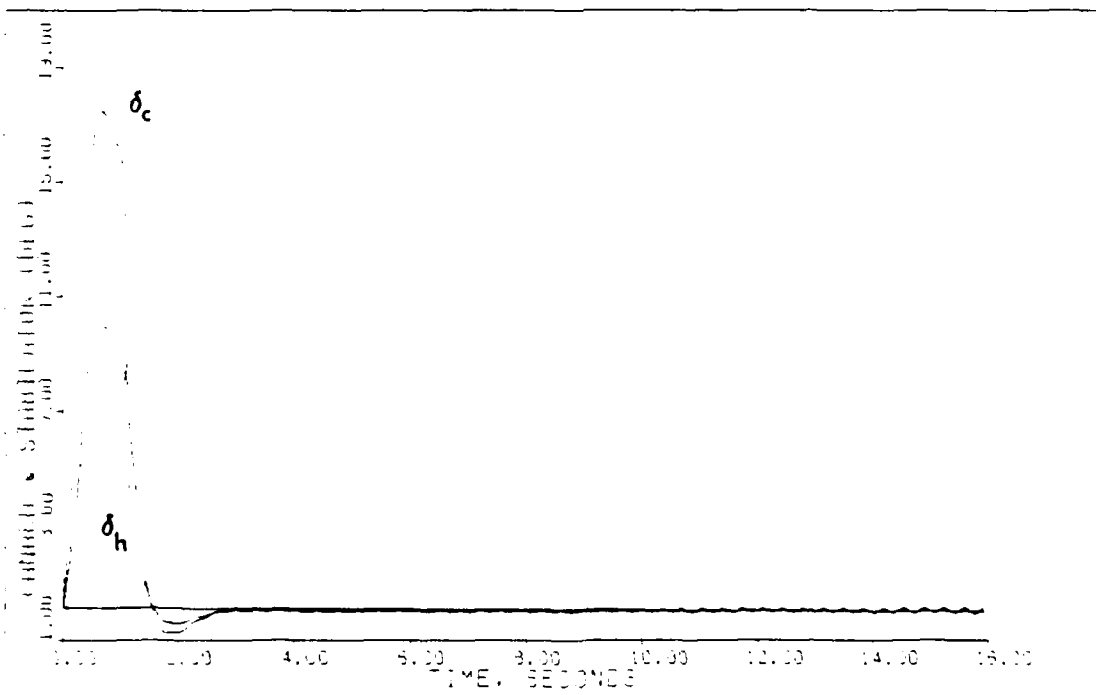
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
0.010	0.285	1.0	.3302E+01	.0000E+00	.5198E+02
		0.9	-.7966E+00	.0000E+00	.3093E+02
		0.50	-.4912E-02	.5700E-02	.9086E-01

Notes:

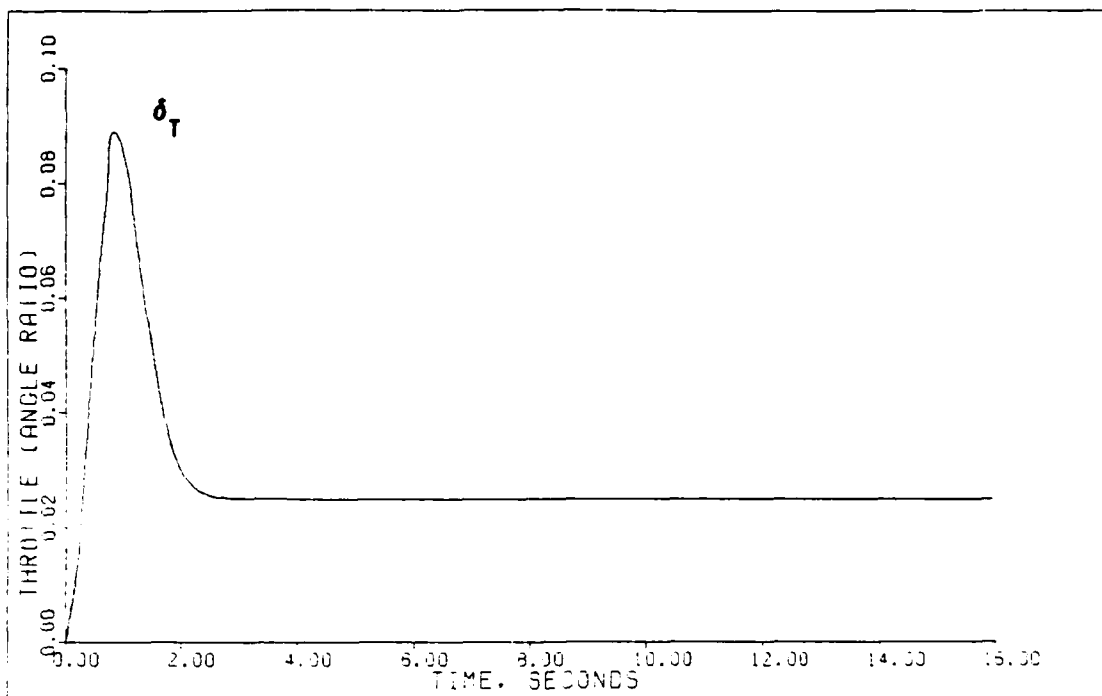
- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\text{alpha})\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



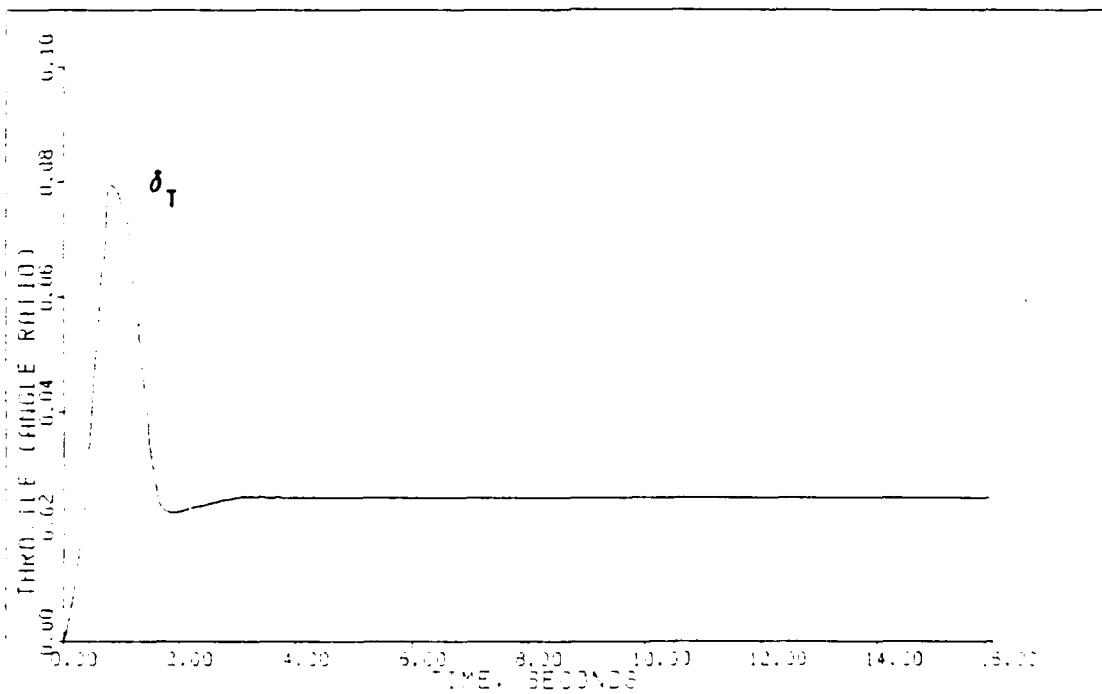
DIRECT CLIMB: PLANT+ACTUATORS (0.9M/F1200)



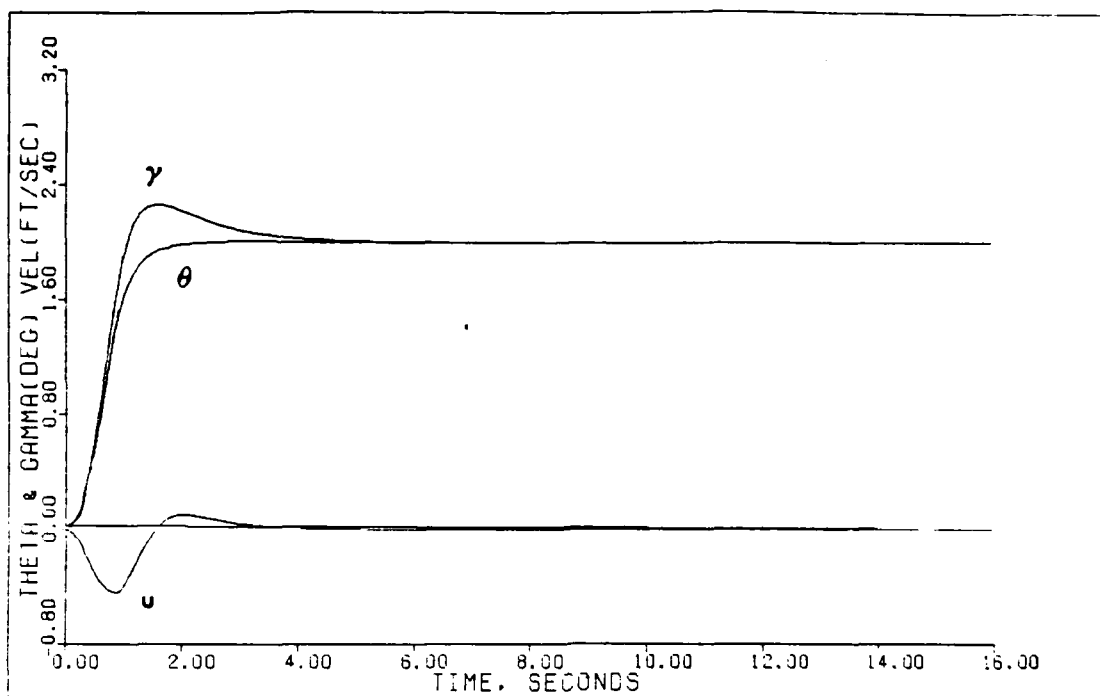
DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSORS (0.9M/F1200)



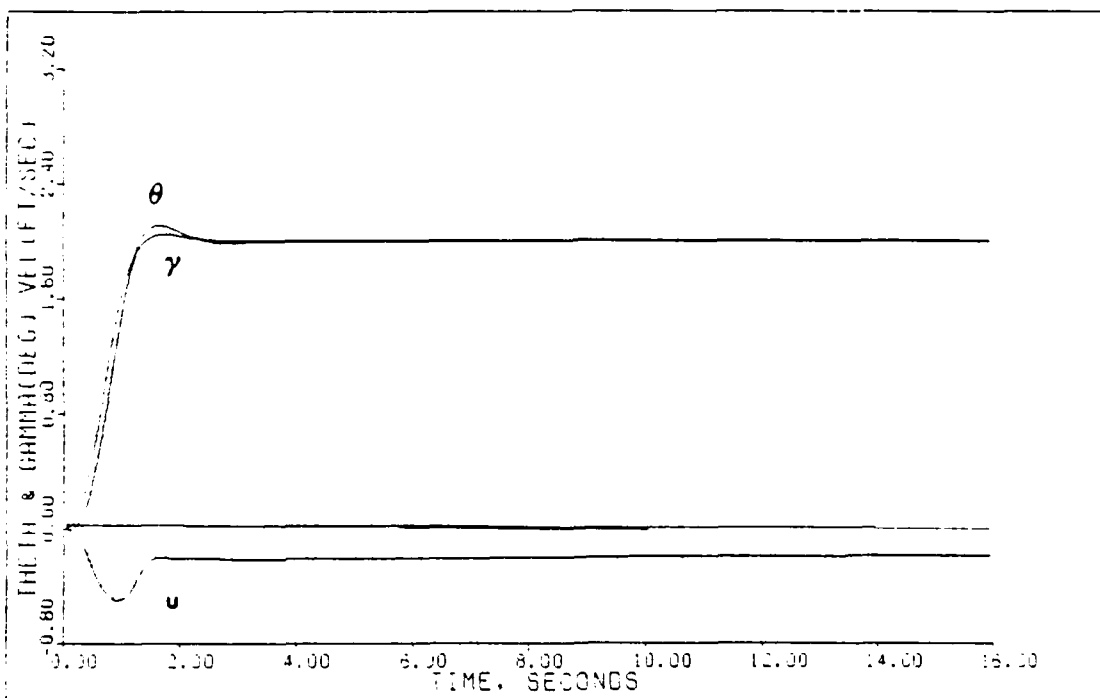
DIRECT CLIMB: PLANT-ACTUATORS (0.9M/FL200)



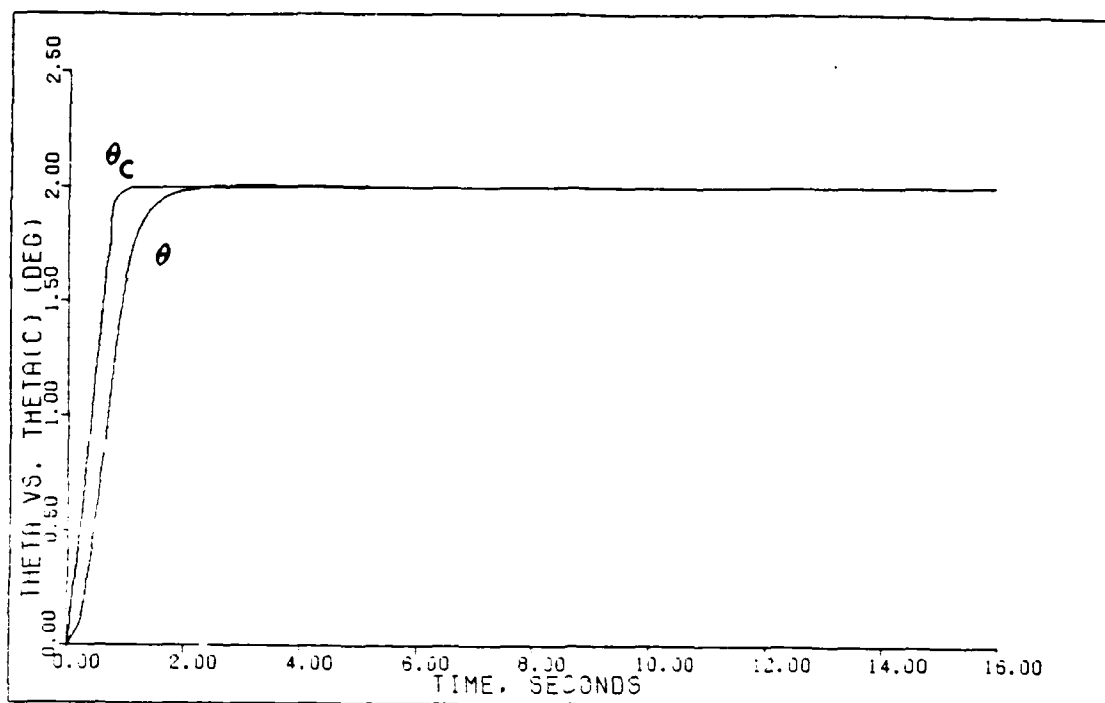
DIRECT CLIMB: PLANT-ACTUATORS+DELAY+SENSORS (0.9M/FL200)



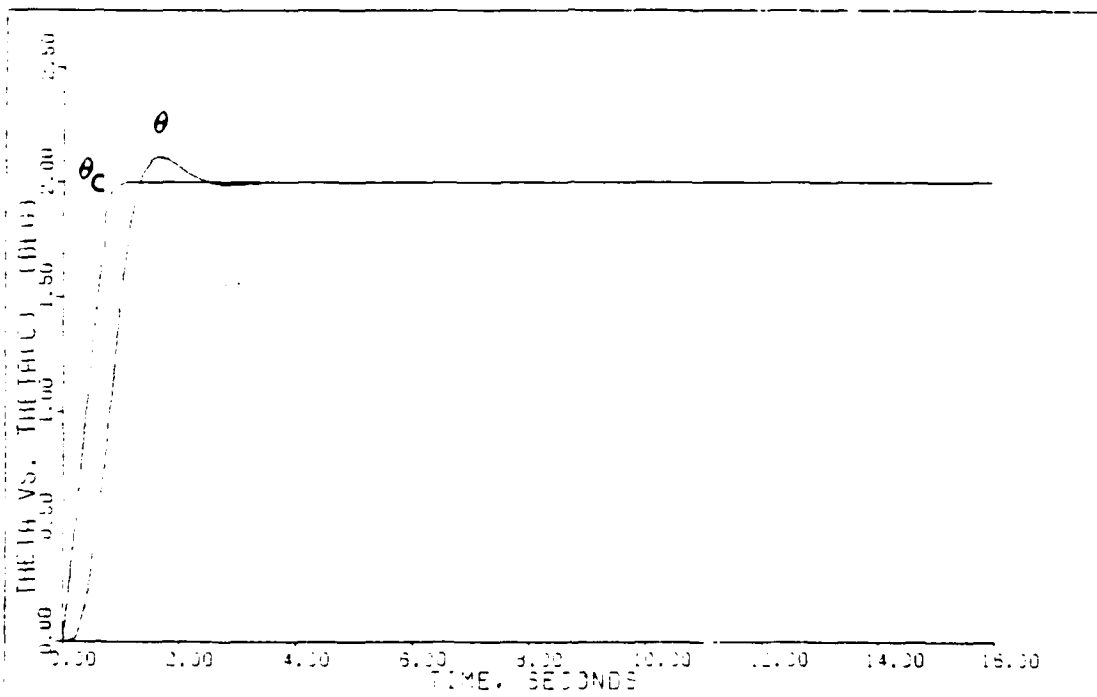
DIRECT CLIMB: PLANT+ACTUATORS (10.9M/FL200)



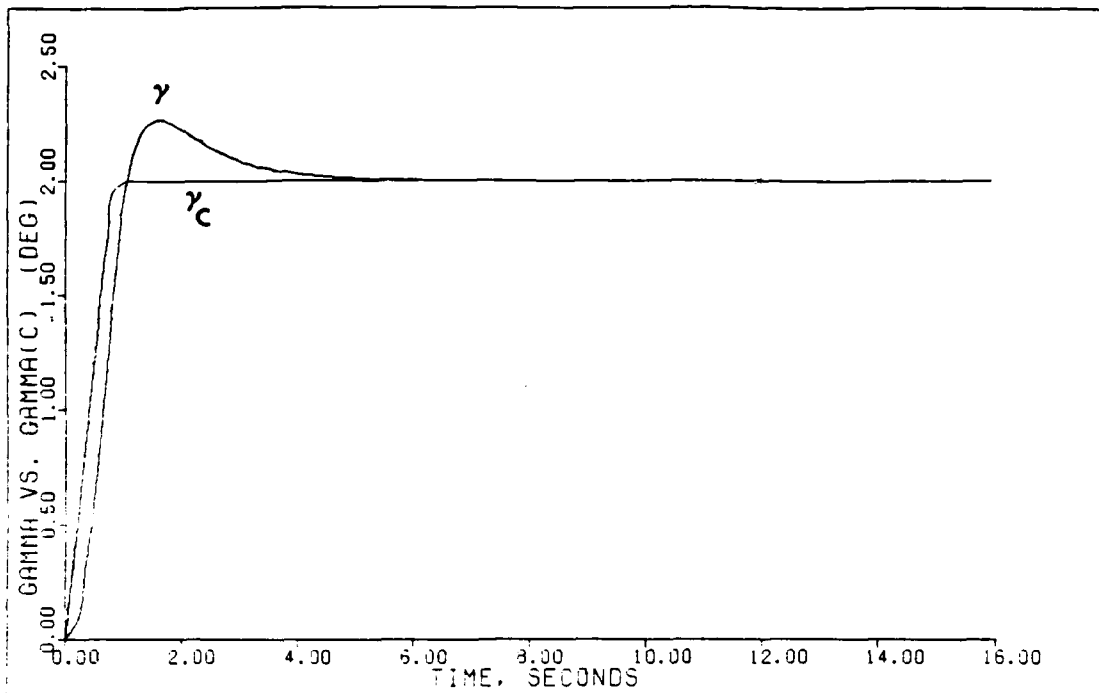
DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSORS (10.9M FL200)



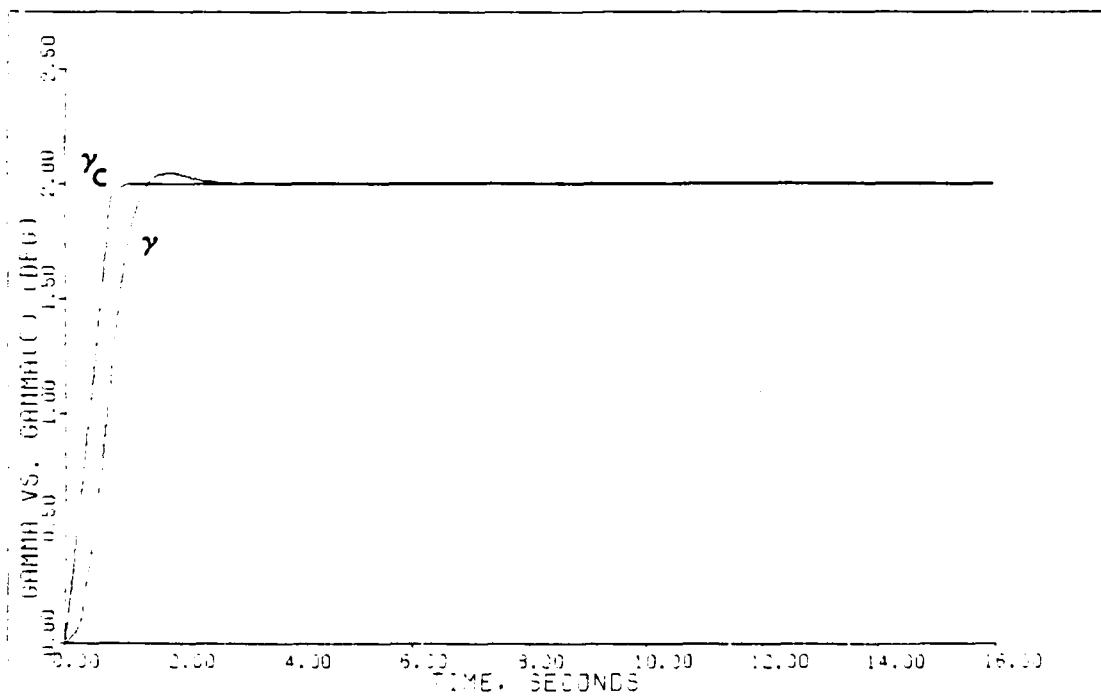
DIRECT CLIMB: PLANT+ACTUATORS (0.9M/FL200)



DIRECT CLIMB: PLANT+ACTUATORS+DELAY+Sensors (0.9M/FL200)



DIRECT CLIMB: PLANT-ACTUATORS (0.9M/FL200)



DIRECT CLIMB: PLANT-ACTUATORS-DELAY-SENSORS (0.9M FL200)

TABLE D.2
DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Direct Climb (+2.0 degs)

Flight Condition: 1.4 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 0.8, 0.03491, 20, 20$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0.8, 0.03491, 20, 20$

Plant + Actuators

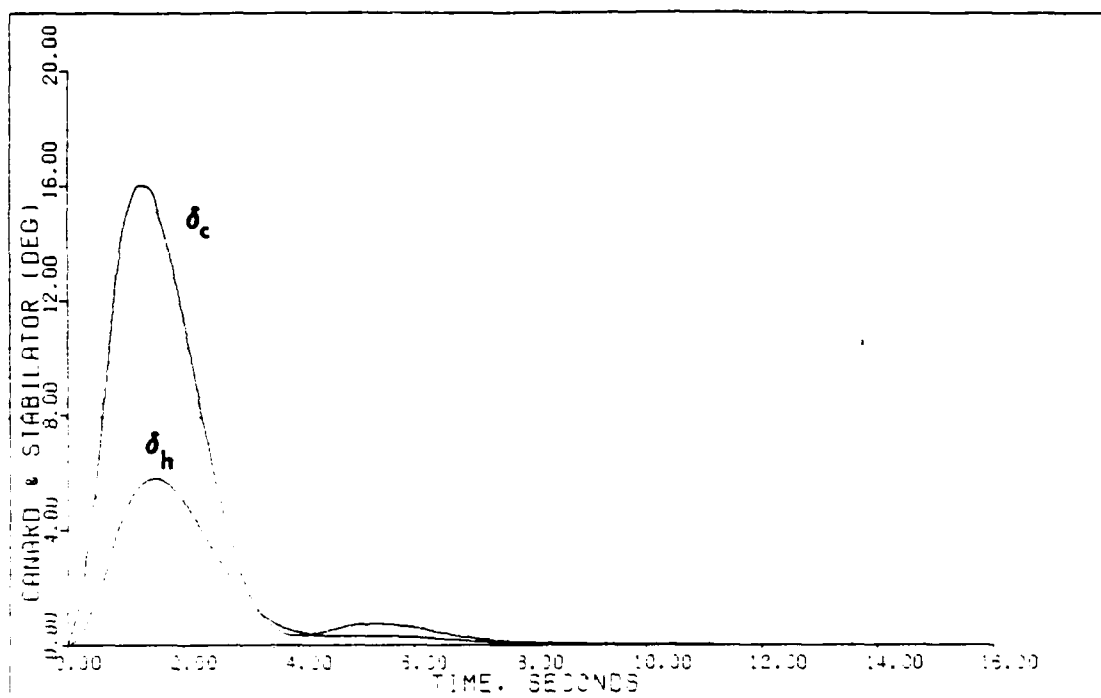
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
10.00	0.120	1.0	.1336E+01	.0000E+00	.1776E+01
		0.3	-.1544E+00	.0000E+00	.7040E+00
		0.025	-.7138E-03	.3000E-03	.2039E-02

Plant + Actuators + Delay + Sensors

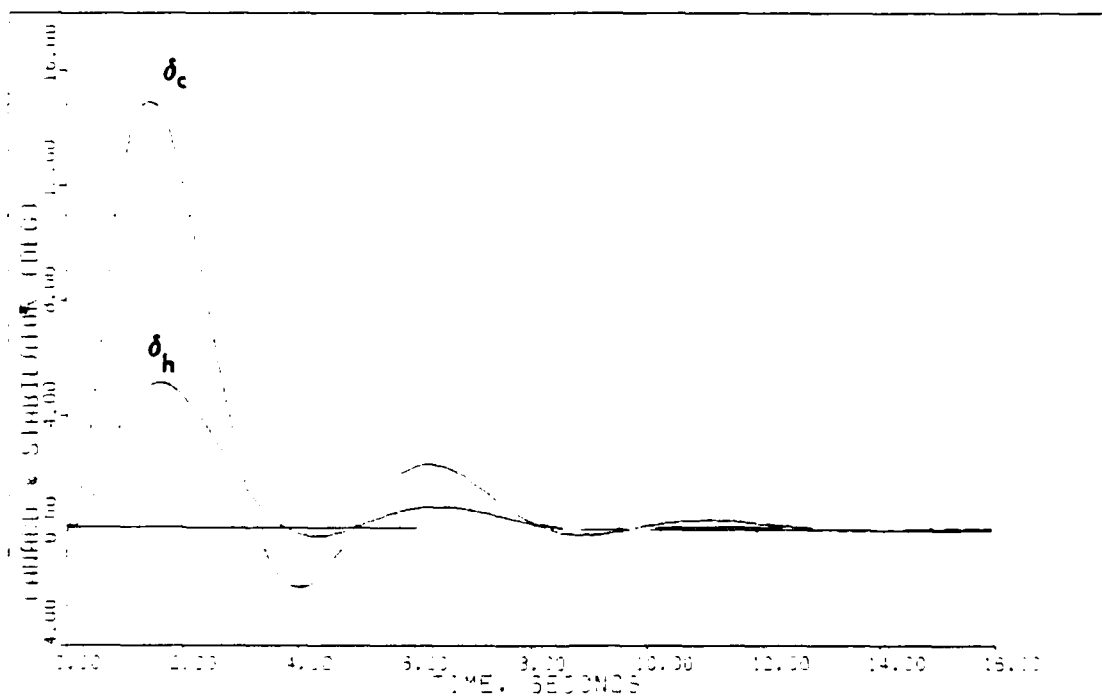
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
5.000	0.180	1.0	.2004E+01	.0000E+00	.4794E+01
		0.9	-.2315E+00	.0000E+00	.1901E+01
		0.045	-.1071E-02	.1350E-02	.5506E-02

Notes:

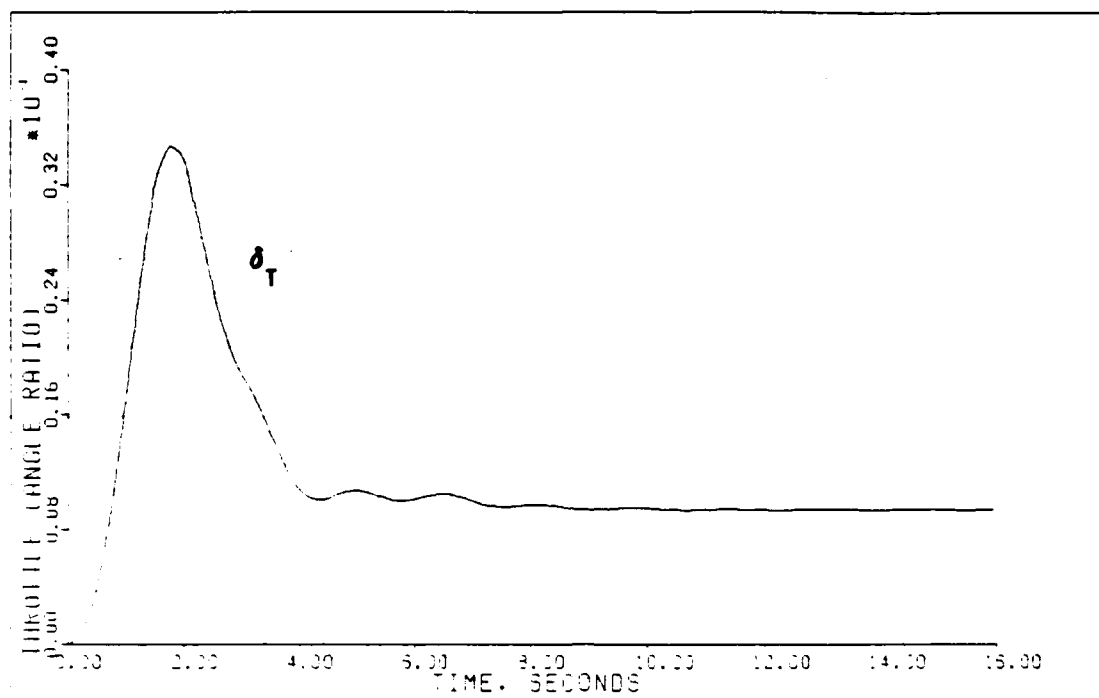
1. Each \underline{v} input is composed of four parts:
 - A. Time (secs) that the input reaches steady-state.
 - B. Steady-state value (radians).
 - C. Time (secs) input leaves steady-state.
 - D. Time (secs) input reaches zero.
2. Sigma = the elements (in order) of the diagonal matrix.
3. The integral controller matrix $\underline{K}_1 = (\text{alpha})\underline{K}_0$.
4. Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



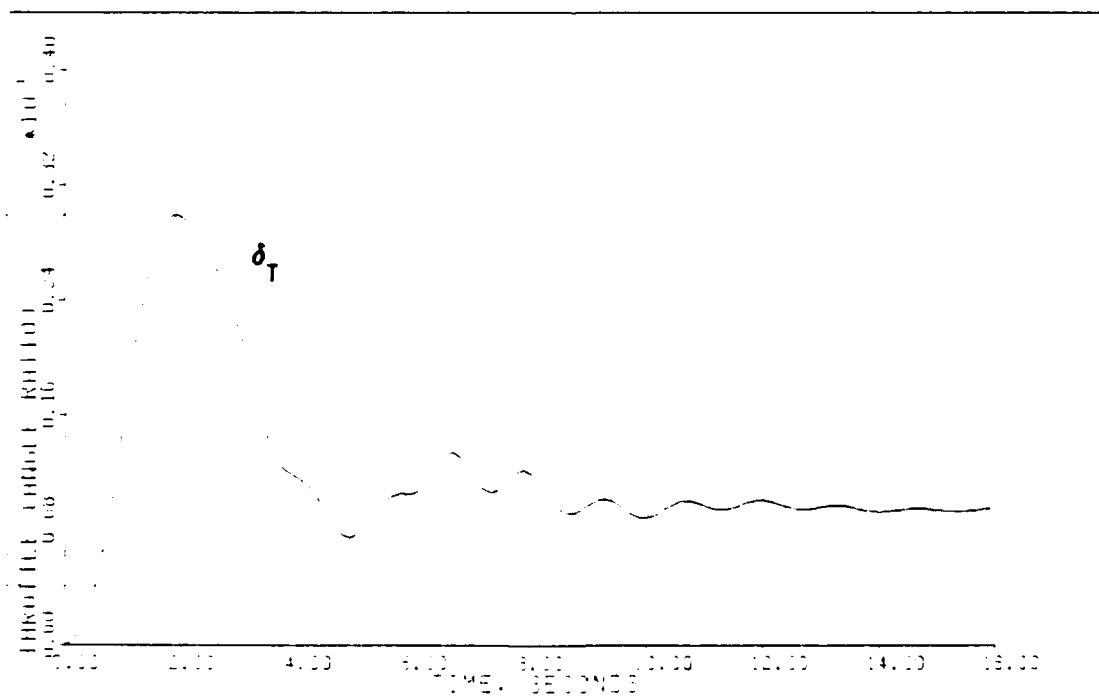
DIRECT CLIMB: PLANT-ACTUATORS (11.4M, FL200)



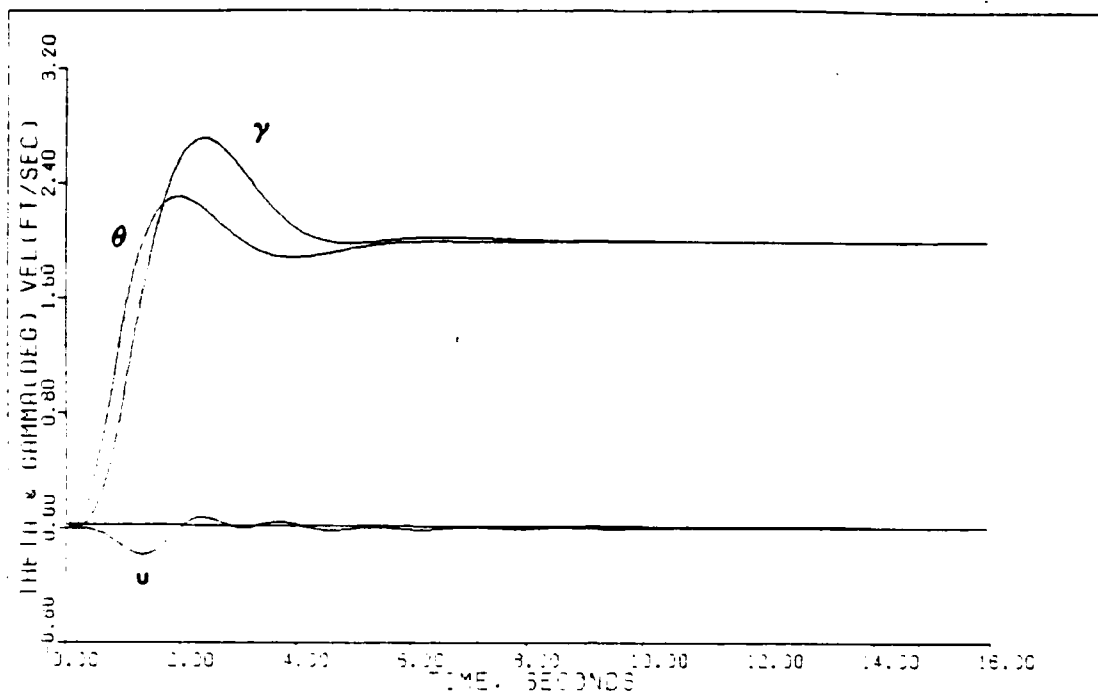
DIRECT CLIMB: PLANT-ACTUATORS-DELAY-SENSORS (11.4M, FL200)



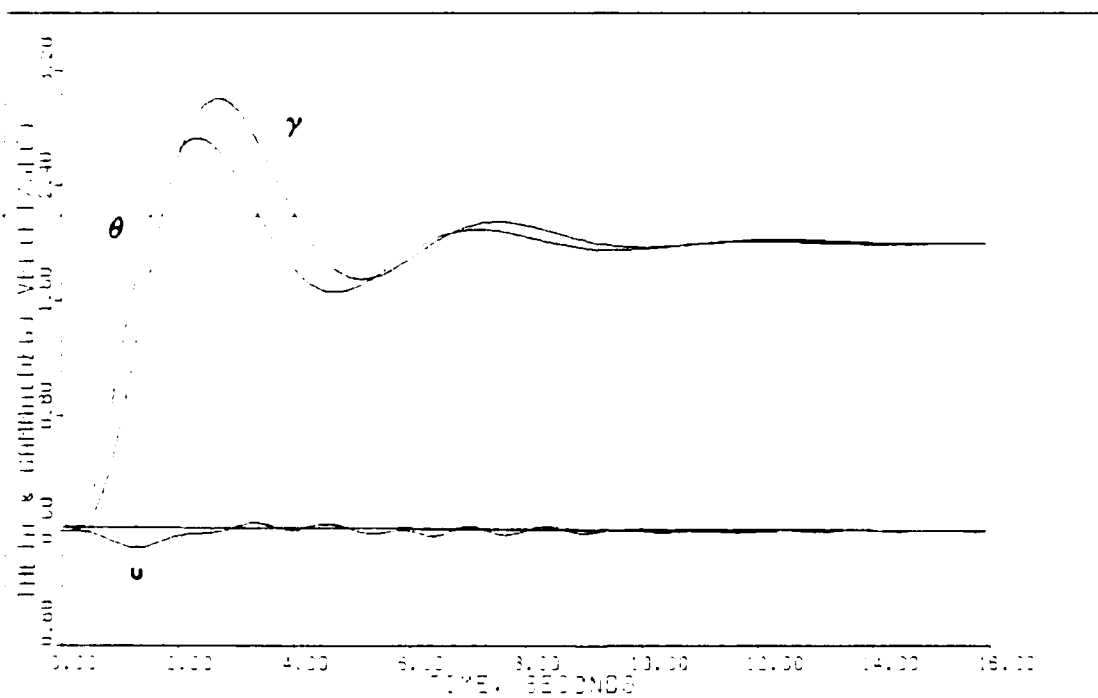
DIRECT CLIMB: PLANT+ACTUATORS 1.4M FL200



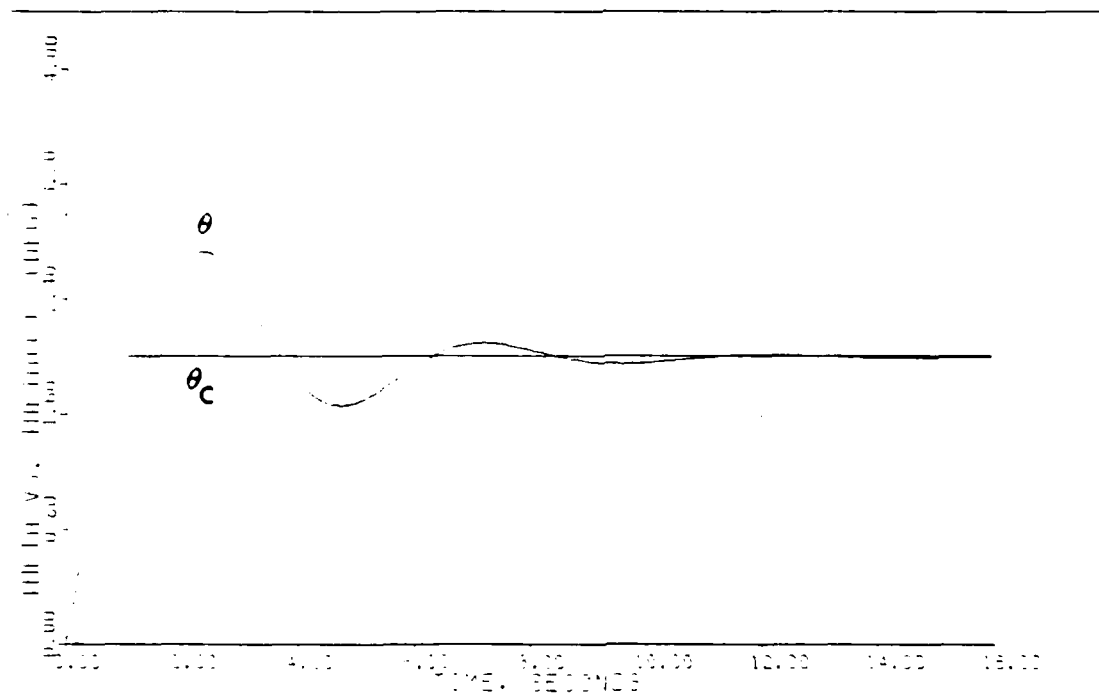
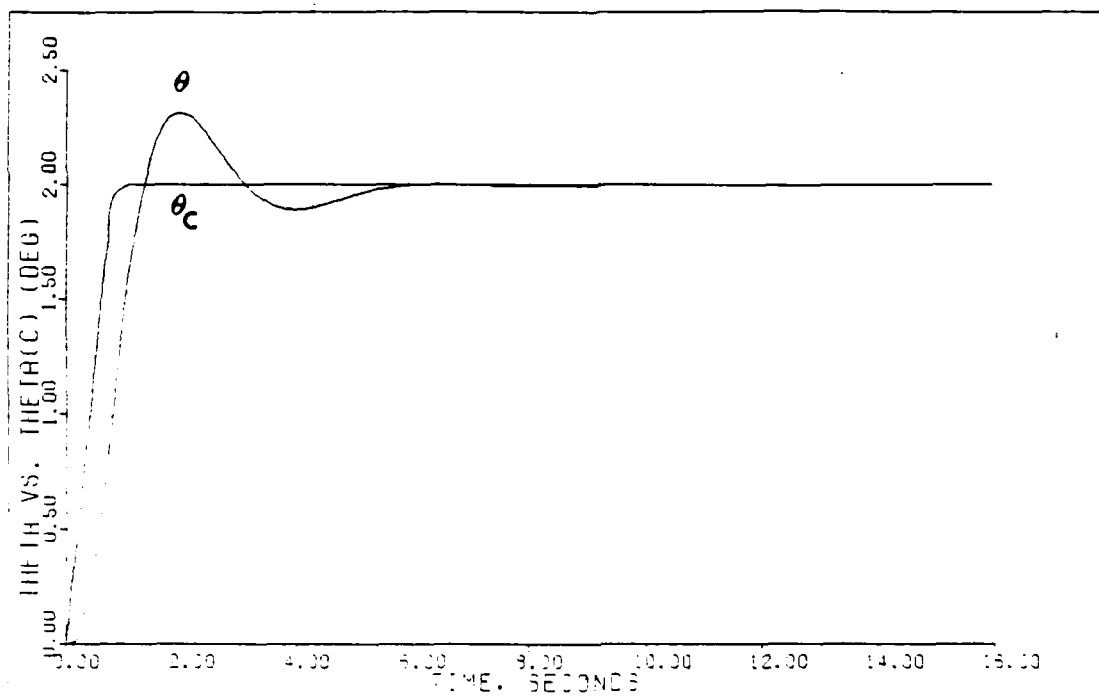
DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSORS 1.4M FL200

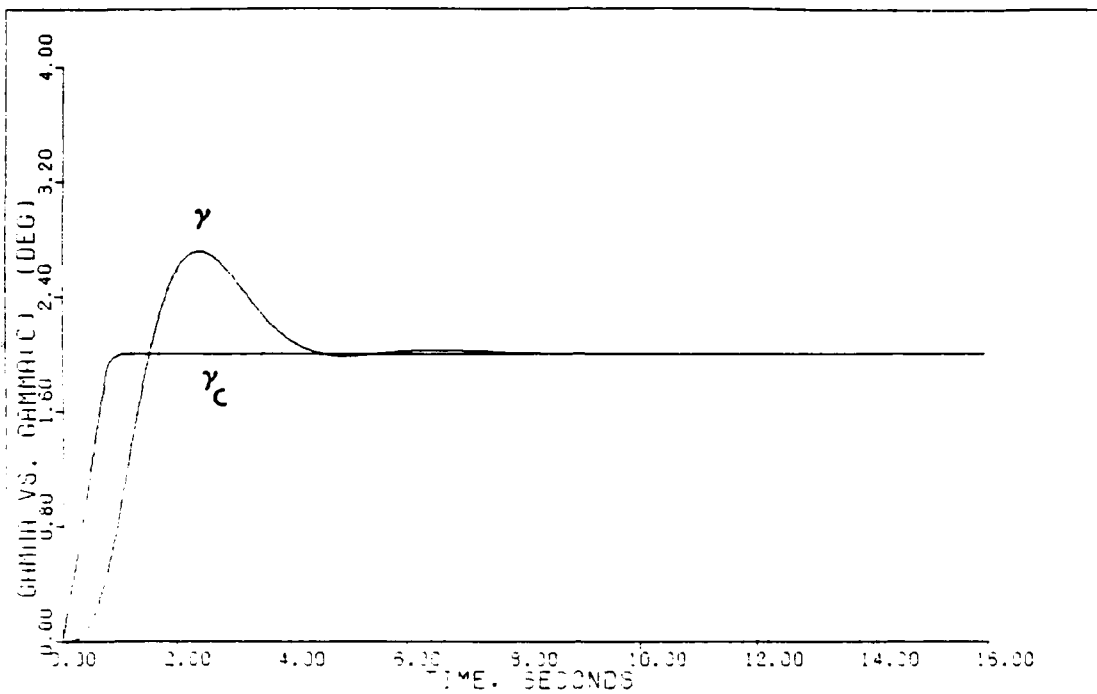


DIRECT CLIMB: PLANT+ACTUATORS 11.4M FL200

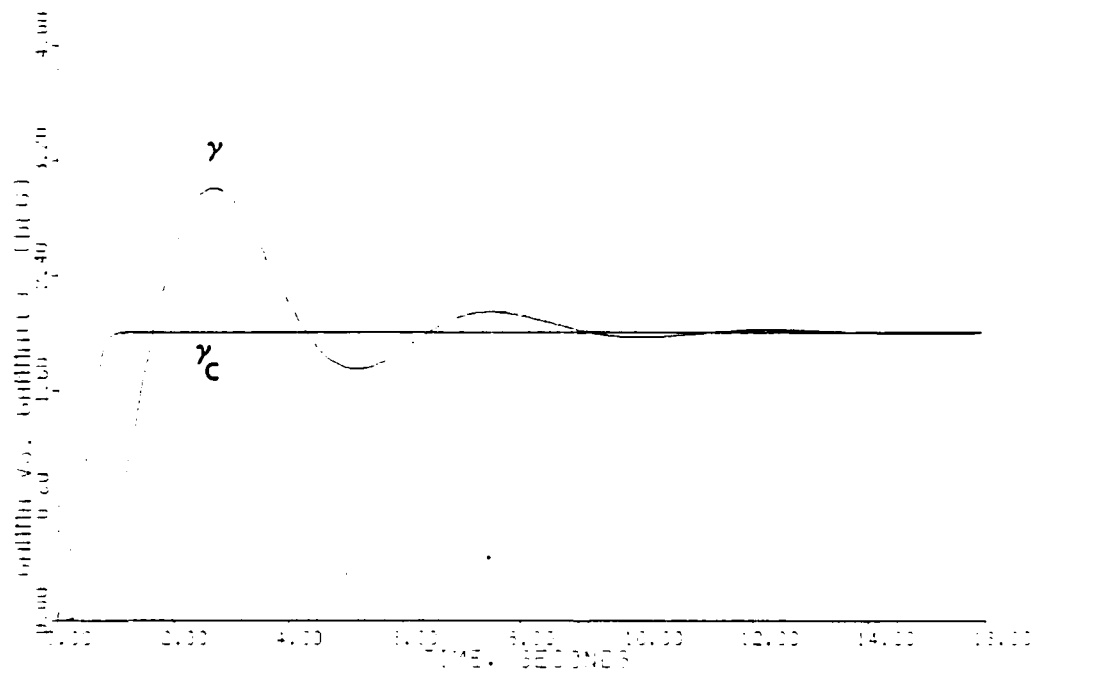


DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSORS 11.4M FL200





DIRECT CLIMB: PLANT+ACTUATORS 1.4M FL2000



DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSOR 1.4M FL2000

TABLE D.3

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Direct Climb (+1.8 degs)

Flight Condition: 2.0 Mach at FL 400

Command Vector \underline{v} : $v_1 = \text{Theta: } 0.8, 0.03142, 20, 20$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0.8, 0.03142, 20, 20$

Plant + Actuators

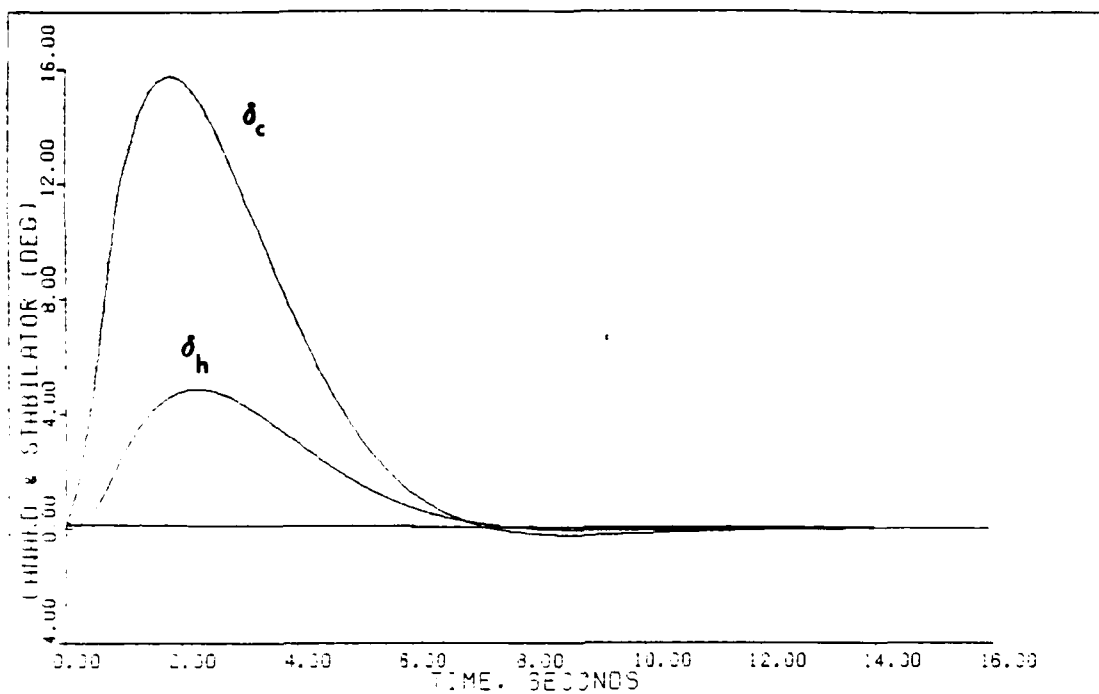
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>K₀</u>		
5.00	0.190	1.0	.2665E+01	.0000E+00	.2473E+01
		0.9	-.8230E+00	.0000E+00	.8290E+00
		0.010	-.1295E-02	.1710E-02	.3032E-02

Plant + Actuators + Delay + Sensors

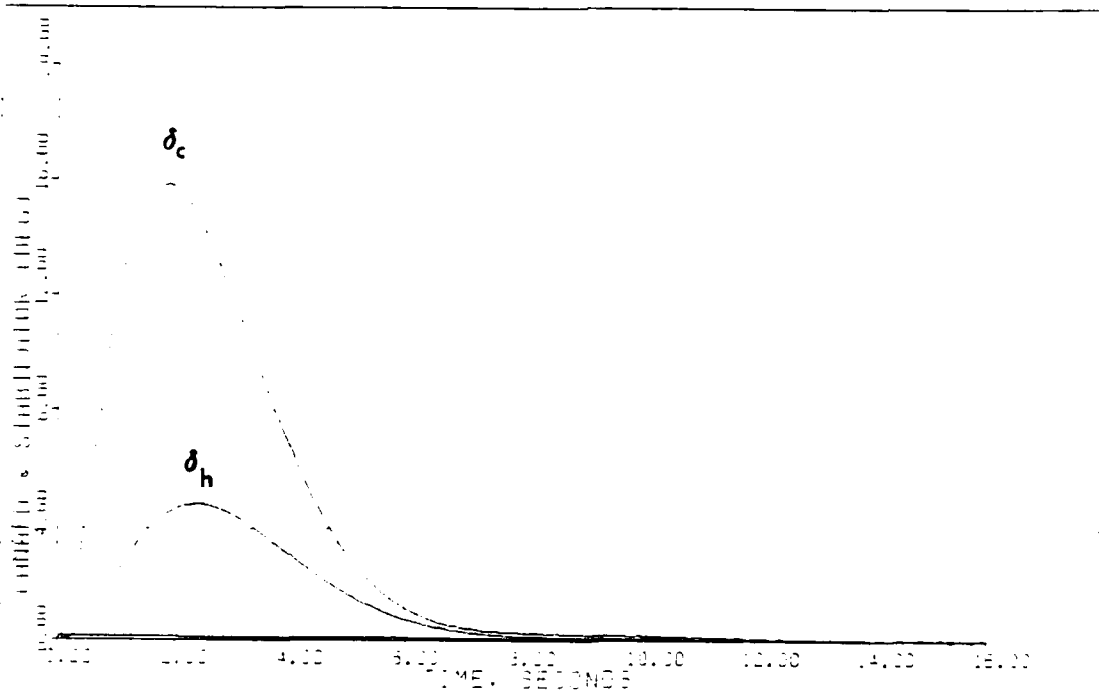
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>K₀</u>		
5.000	0.190	1.0	.2665E+01	.0000E+00	.4204E+01
		0.9	-.8230E+00	.0000E+00	.1409E+01
		0.017	-.1295E-02	.1710E-02	.5154E-02

Notes:

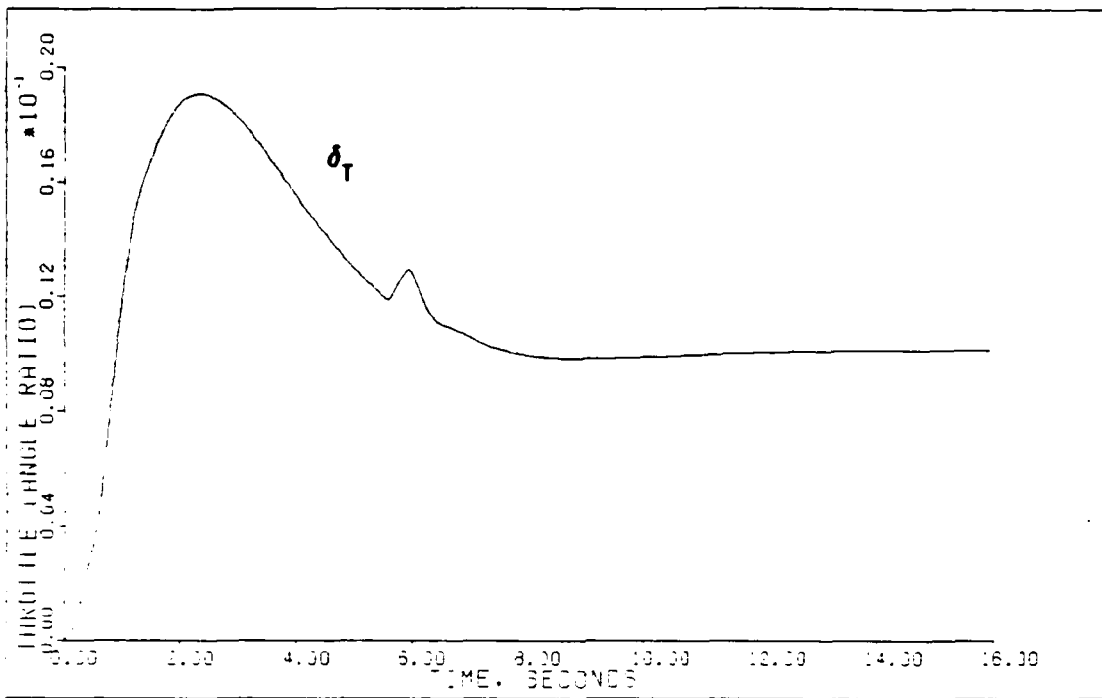
- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\alpha)\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



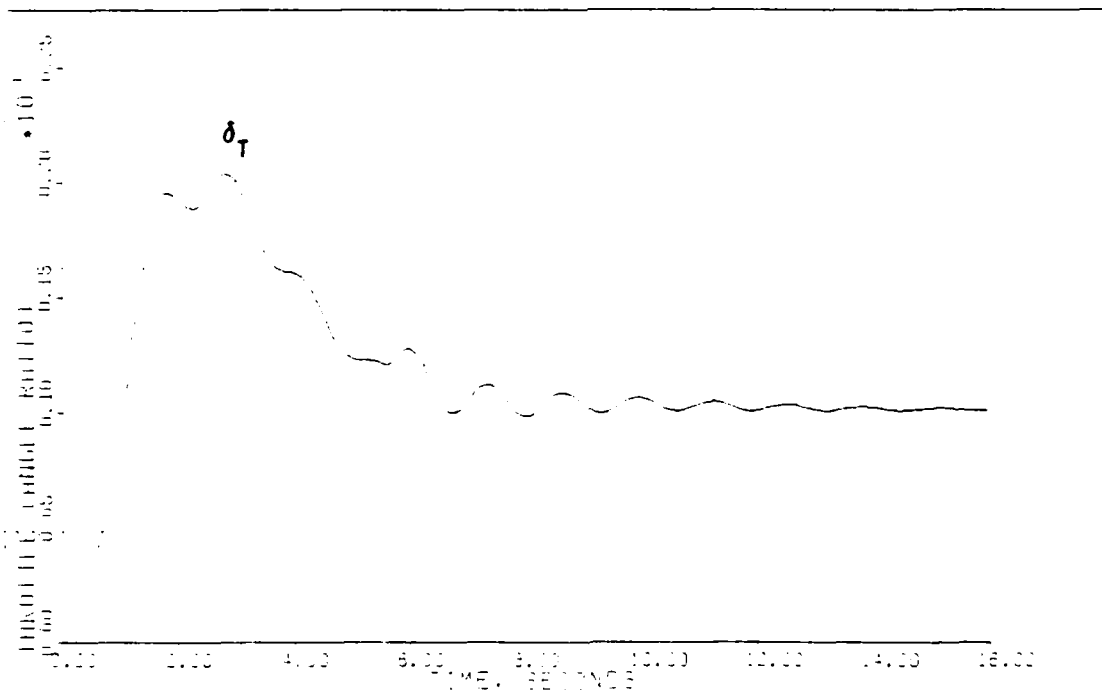
DIRECT CLIMB: PLANT+ACTUATORS (2.0M/FL400)



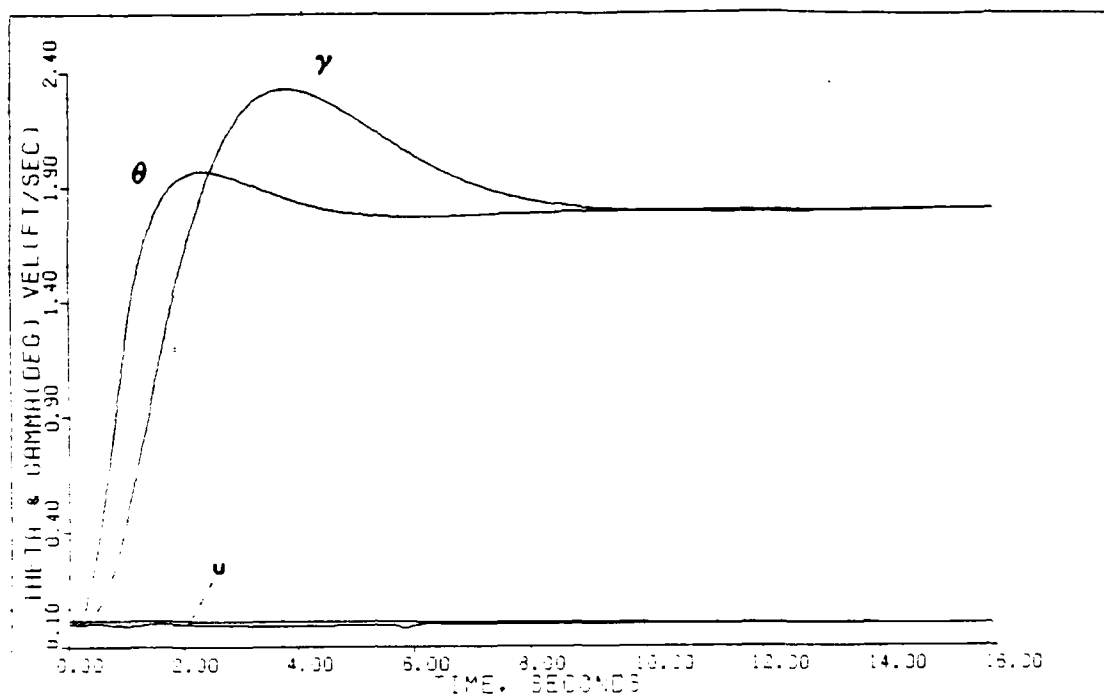
DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSORS (2.0M/FL400)



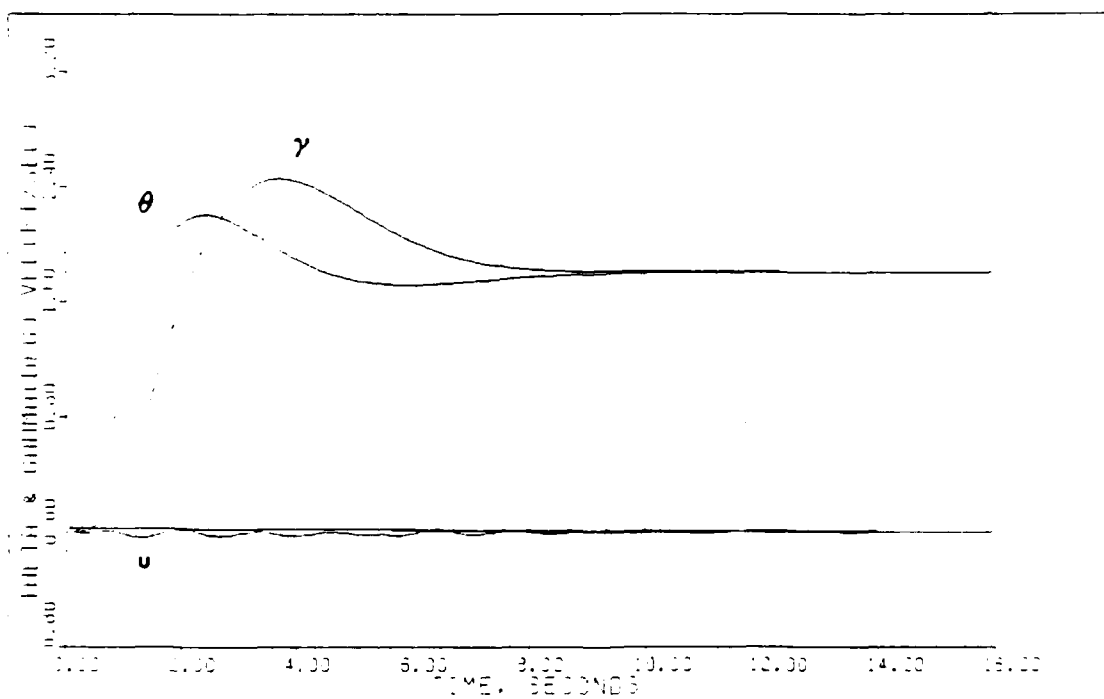
DIRECT CLIMB: PLANT+ACTUATORS 2.0M FL400



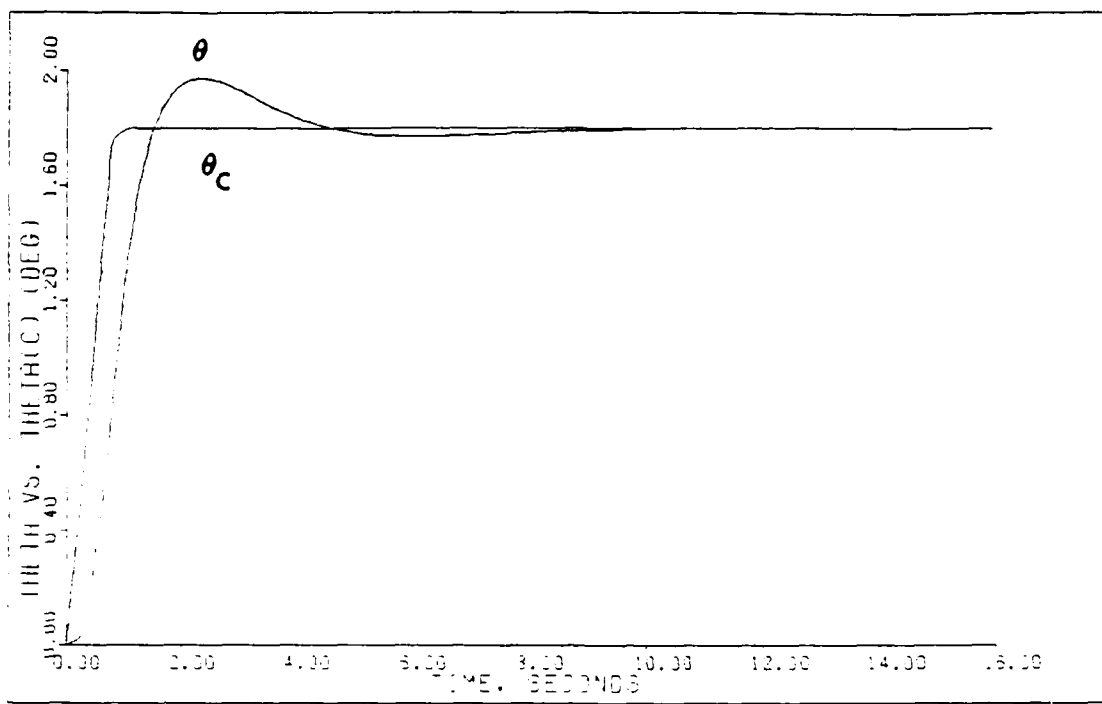
DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSORS 2.0M FL400



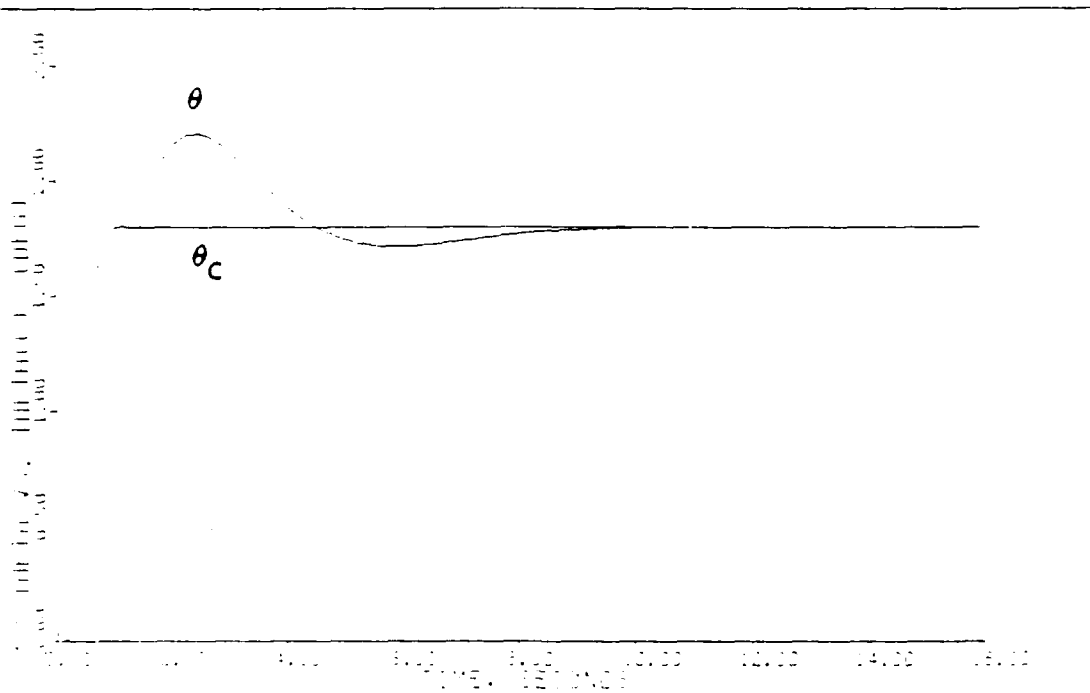
DIRECT CLIMB: PLANT+ACTUATORS (2.0M FL400)



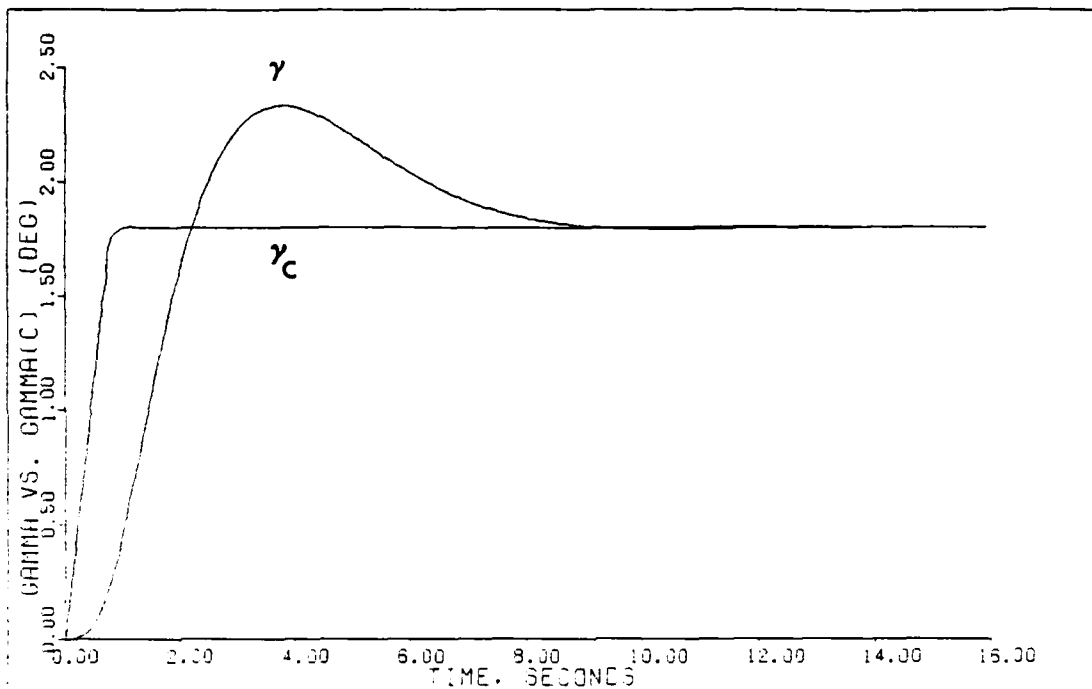
DIRECT CLIMB: PLANT+ACTUATORS+DELAY+SENSORS (2.0M FL400)



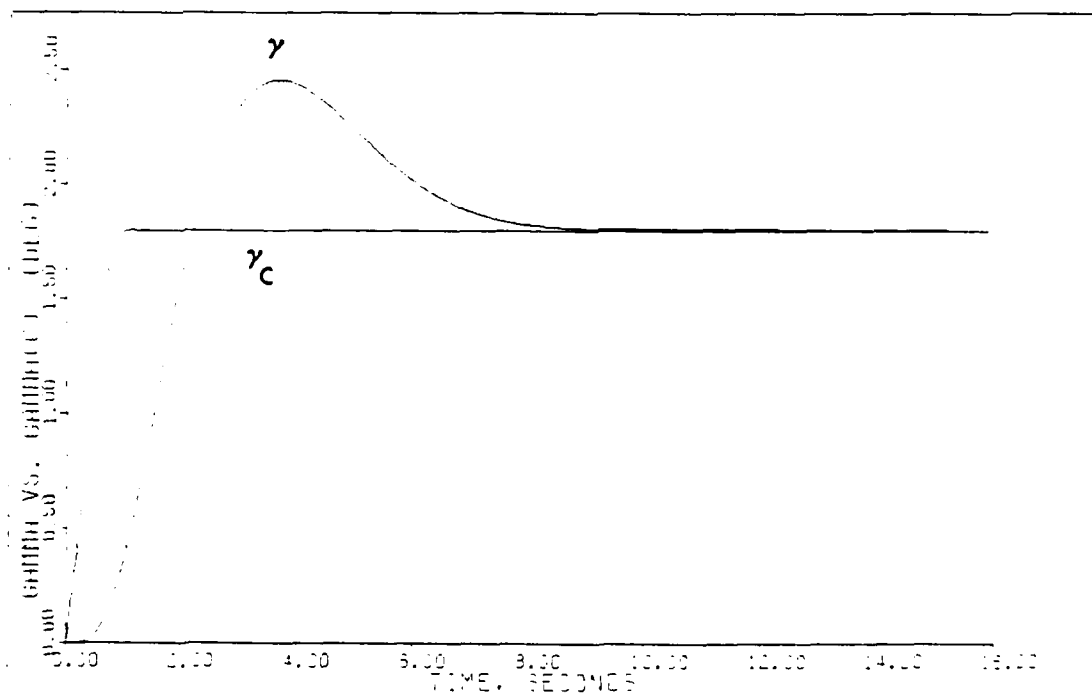
DIRECT CLIMB: PLANT-ACTUATORS 2.0M FL400



DIRECT CLIMB: PLANT-ACTUATOR-DELTA-SENSOR 2.0M FL400



DIRECT CLIMB: PLANT-ACTUATORS (2.0M/FL400)



DIRECT CLIMB: PLANT-ACTUATORS-DELAY-SENSORS (2.0M/FL400)

TABLE D.4

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Vertical Translation (+2.0 degs)

Flight Condition: 0.3 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 0, 0, 0, 0$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0.8, 0.03490, 20, 20$

Plant + Actuators

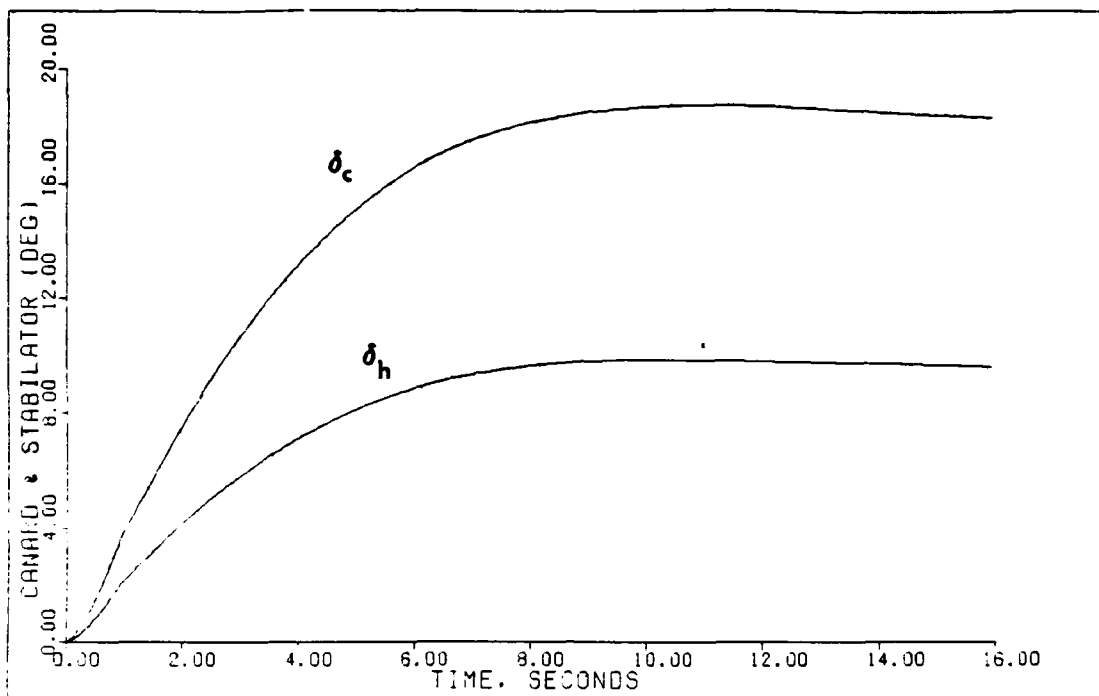
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
		1.0	.4273E+02	.0000E+00	.1179E+01
2.50	0.400	1.0	-.1204E+02	.0000E+00	.6685E+00
		0.0028	-.2840E-01	.2000E+01	.3115E-02

Plant + Actuators + Delay + Sensors

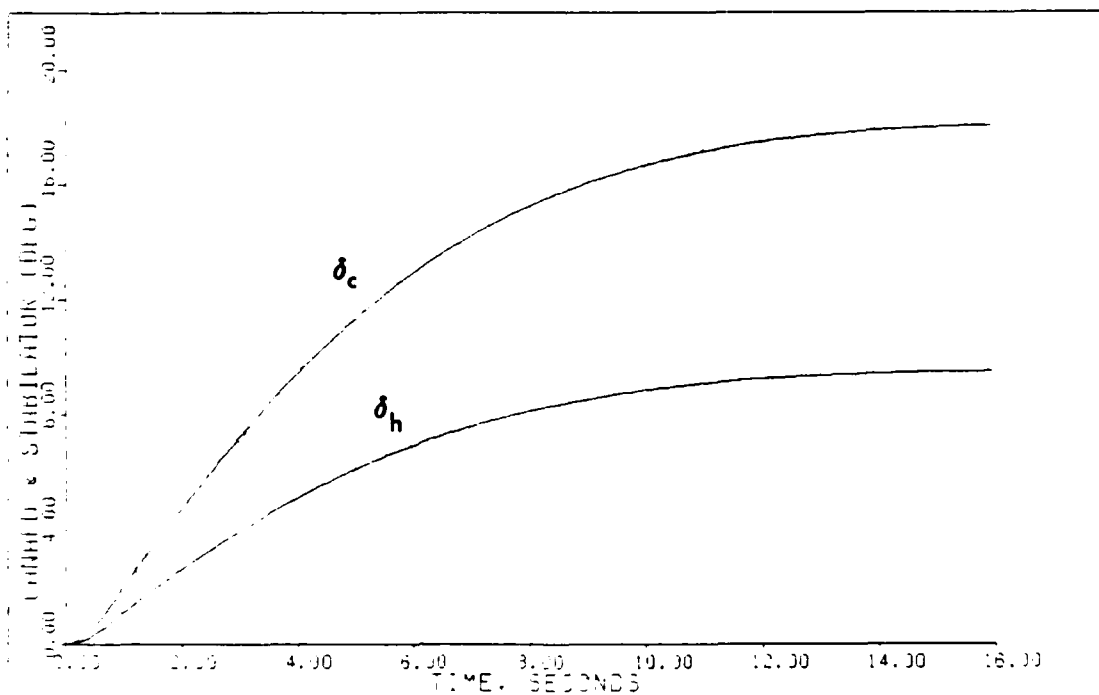
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
		1.0	.2136E+02	.0000E+00	.8003E+00
5.000	0.200	1.0	-.6022E+01	.0000E+00	.4536E+00
		0.0038	-.1420E-01	.1000E-01	.2113E-02

Notes:

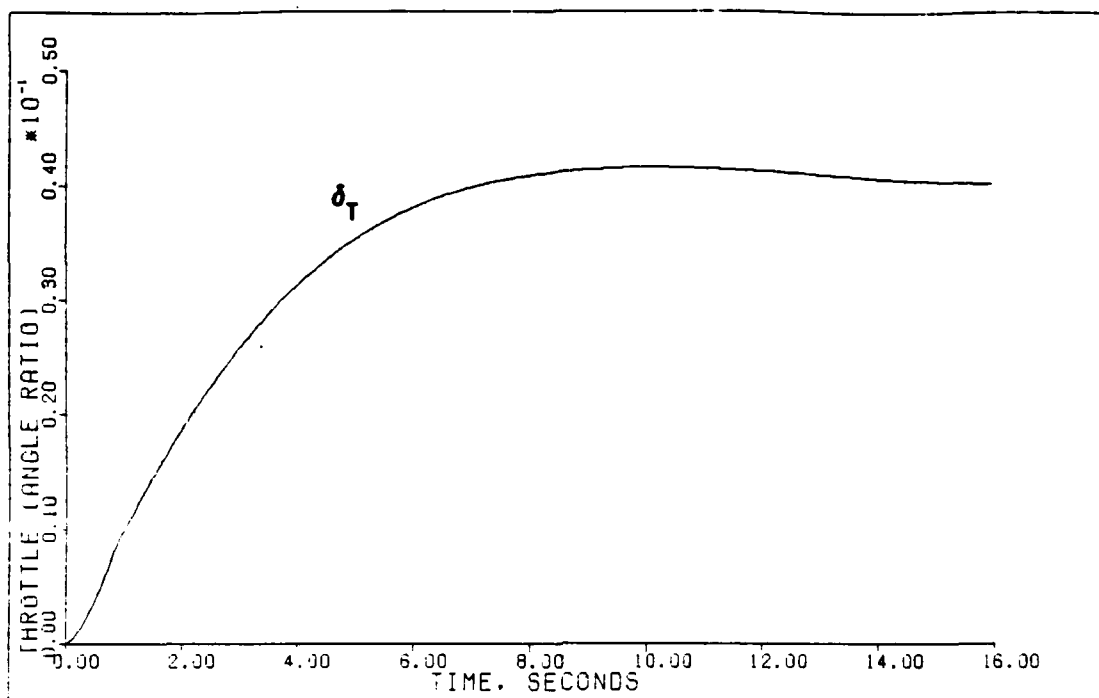
- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\alpha)\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



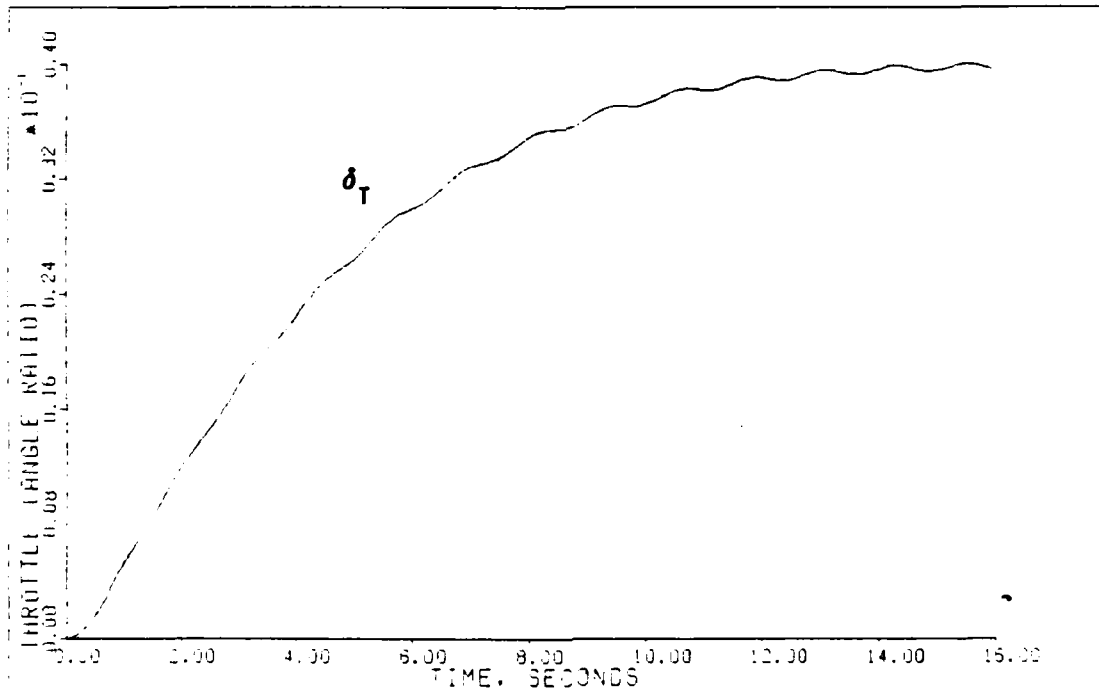
VERTICAL TRANS: PLANT+ACTUATORS (0.3M/FL200)



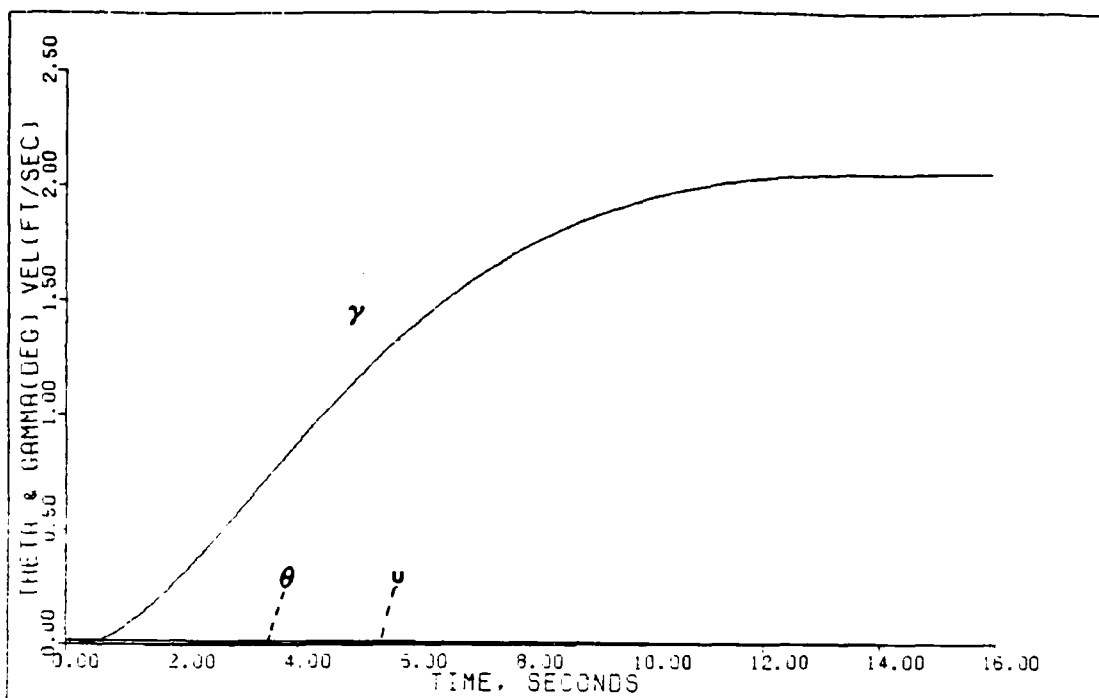
VERTICAL TRANS: PLANT+ACTUATORS+DELAY+SENSORS (0.3M/FL200)



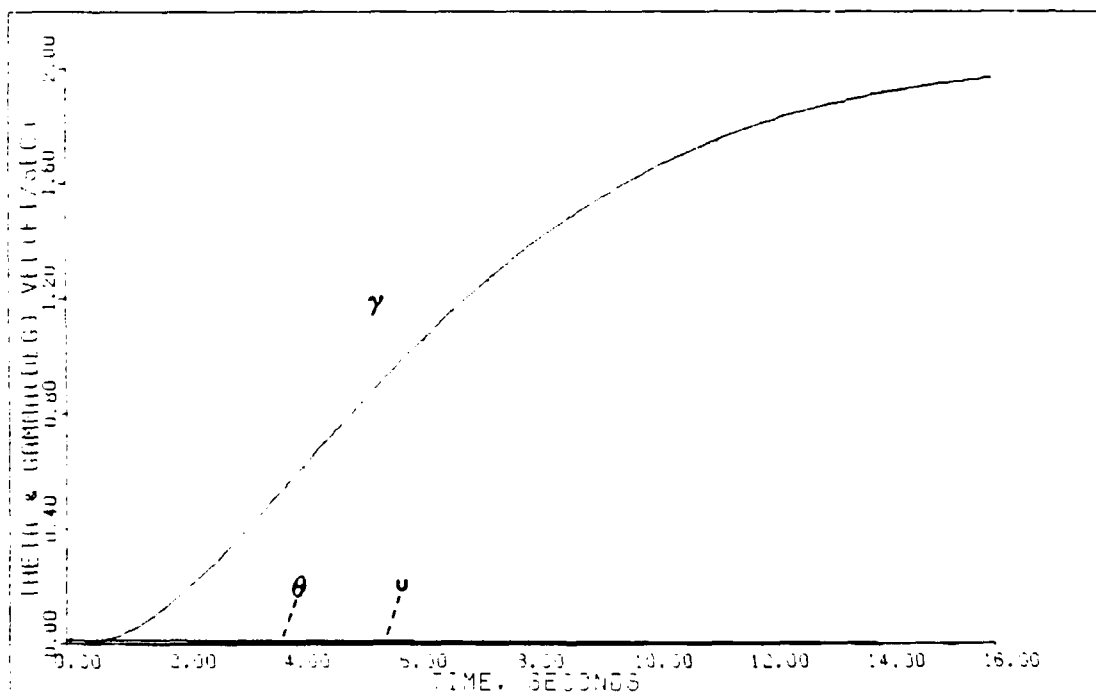
VERTICAL TRANS: PLANT+ACTUATORS (0.3M/FL200)



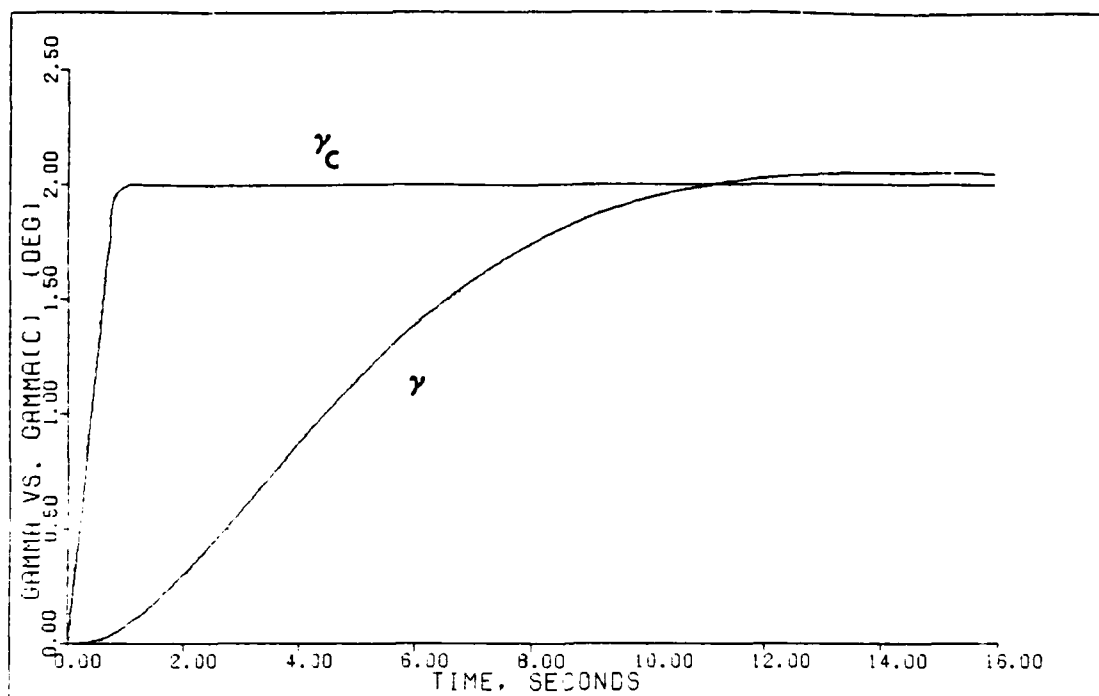
VERTICAL TRANS: PLANT+ACTUATORS+DELAY+SENSORS (0.3M/FL200)



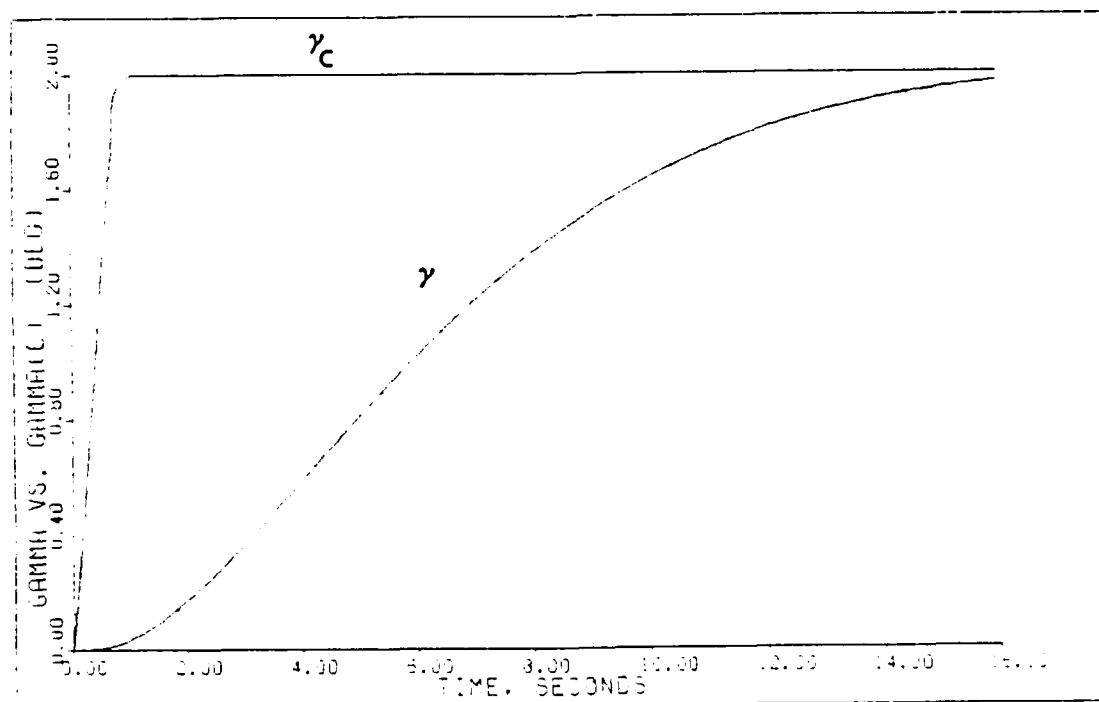
VERTICAL TRANS: PLANT+ACTUATORS (0.3M/FL200)



VERTICAL TRANS: PLANT+ACTUATORS+DELAY+SENSORS (0.3M/FL200)



VERTICAL TRANS: PLANT+ACTUATORS (0.3M/FL200)



VERTICAL TRANS: PLANT+ACTUATORS+DELAY+Sensors (0.3M/FL200)

TABLE D.5

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Vertical Translation (+1.0 deg)

Flight Condition: 1.4 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 0, 0, 0, 0$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0.8, 0.01745, 20, 20$

Plant + Actuators

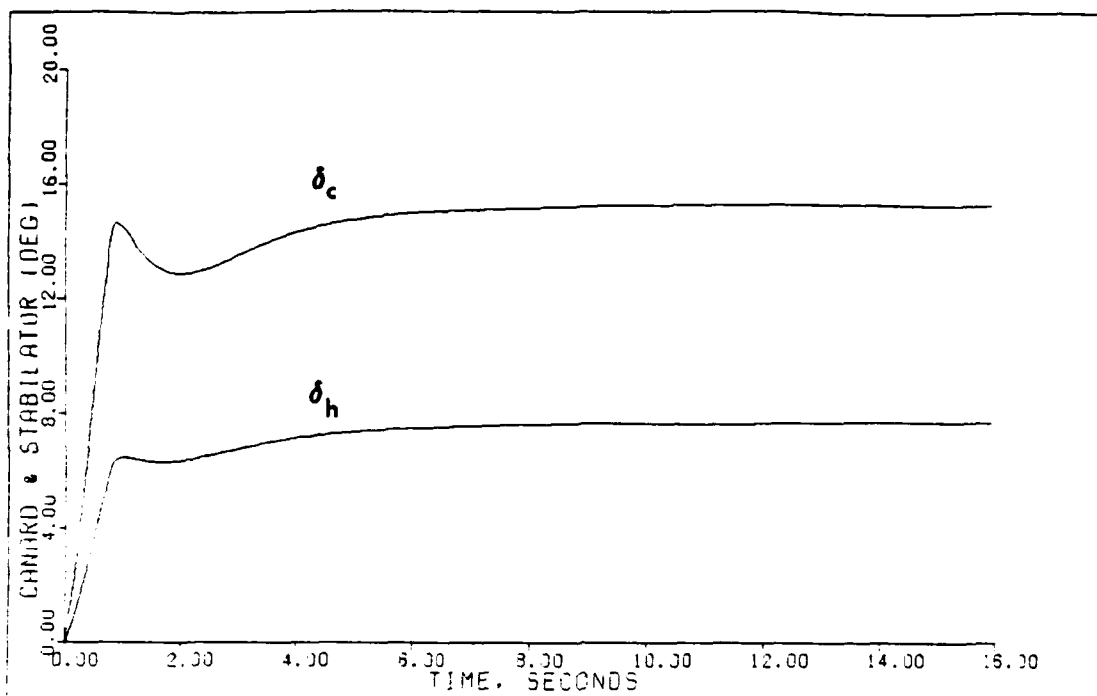
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
1.428	0.770	1.0	.8574E+01	.0000E+00	.2279E+02
		1.0	-.9905E+00	.0000E+00	.9034E+01
		0.05	-.4580E-02	.6417E-02	.2617E-01

Plant + Actuators + Delay + Sensors

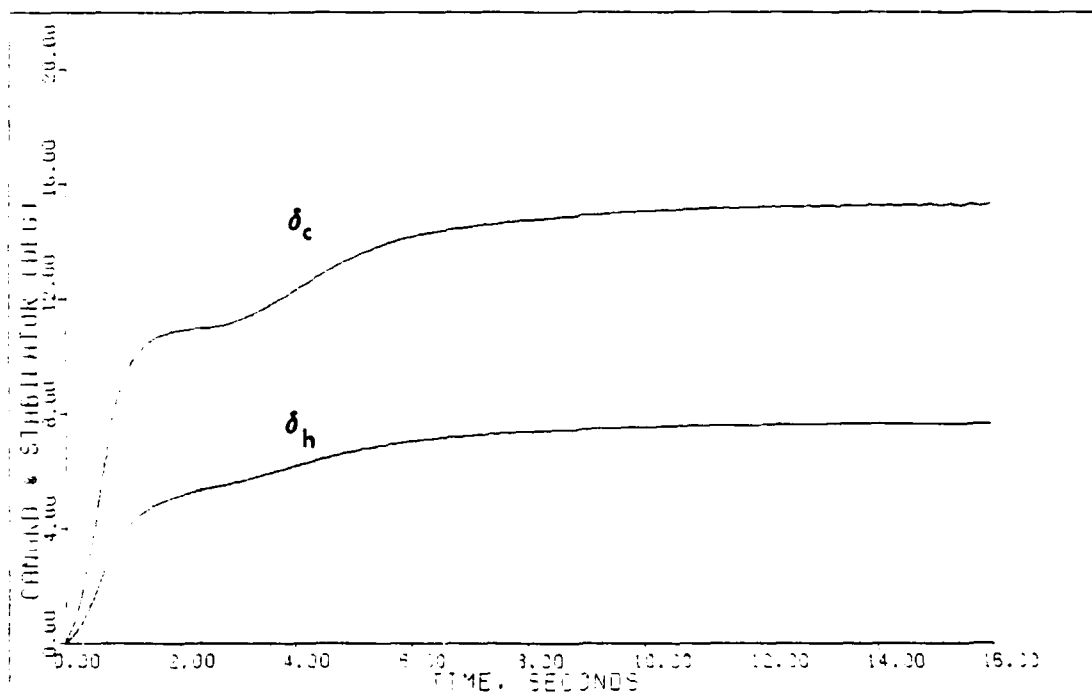
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
2.000	0.550	1.0	.6125E+01	.0000E+00	.1628E+02
		0.5	-.7075E+00	.0000E+00	.6453E+01
		0.05	-.3271E-02	.2292E-02	.1869E-01

Notes:

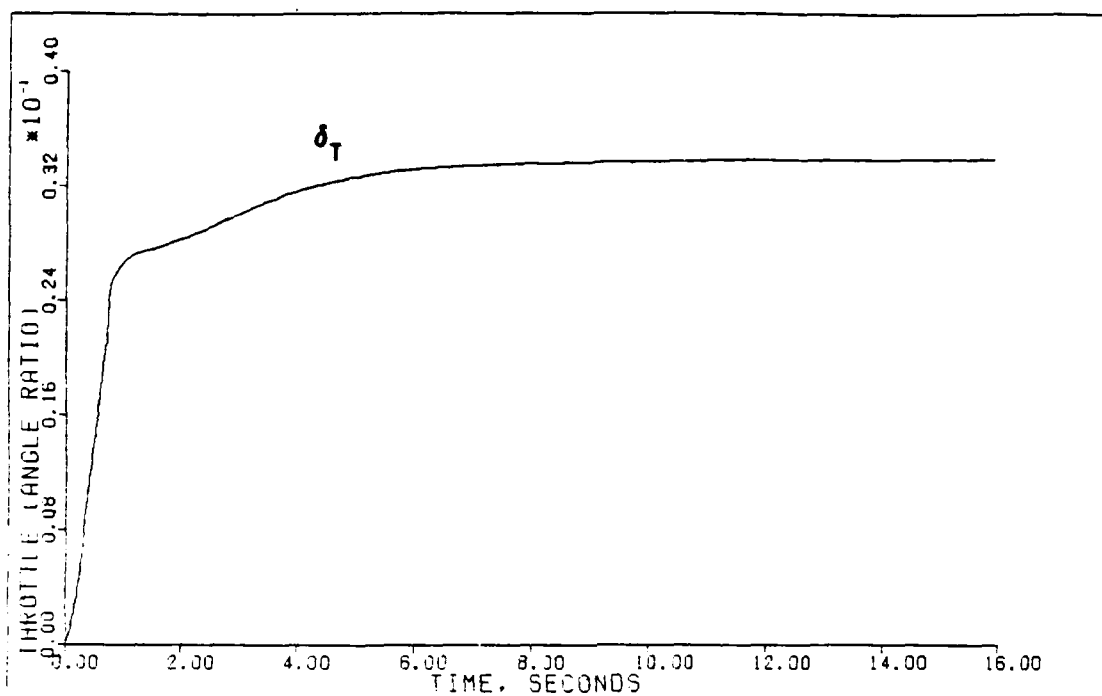
- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\alpha)\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



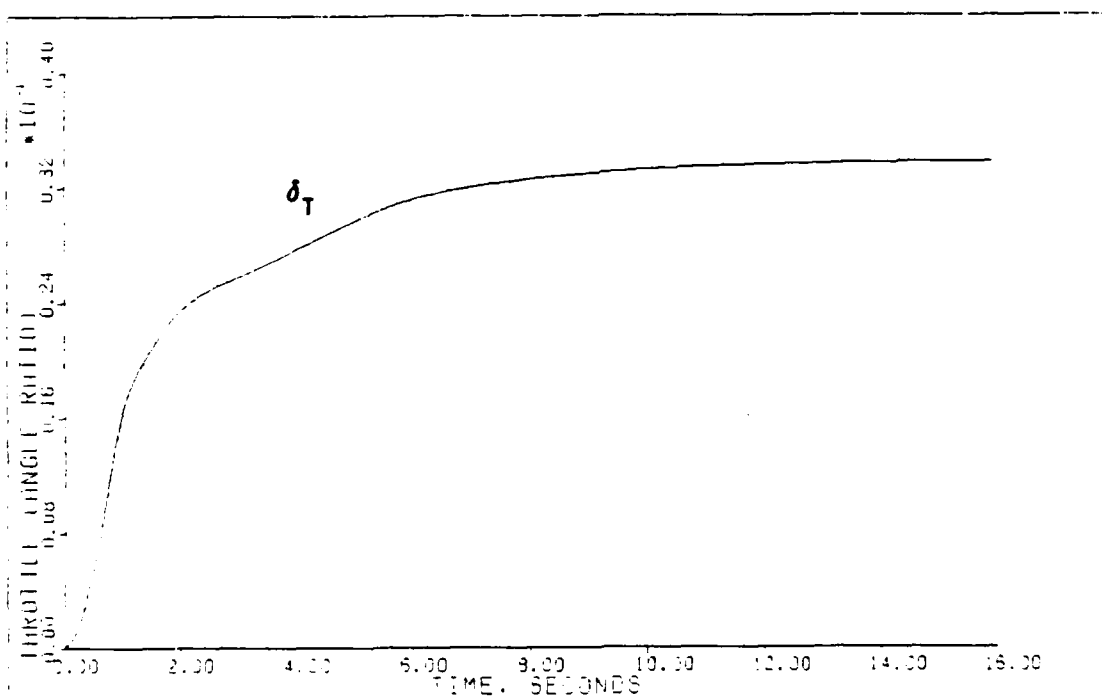
VERTICAL TRANS: PLANT-ACTUATORS (1.4M FLOOD)



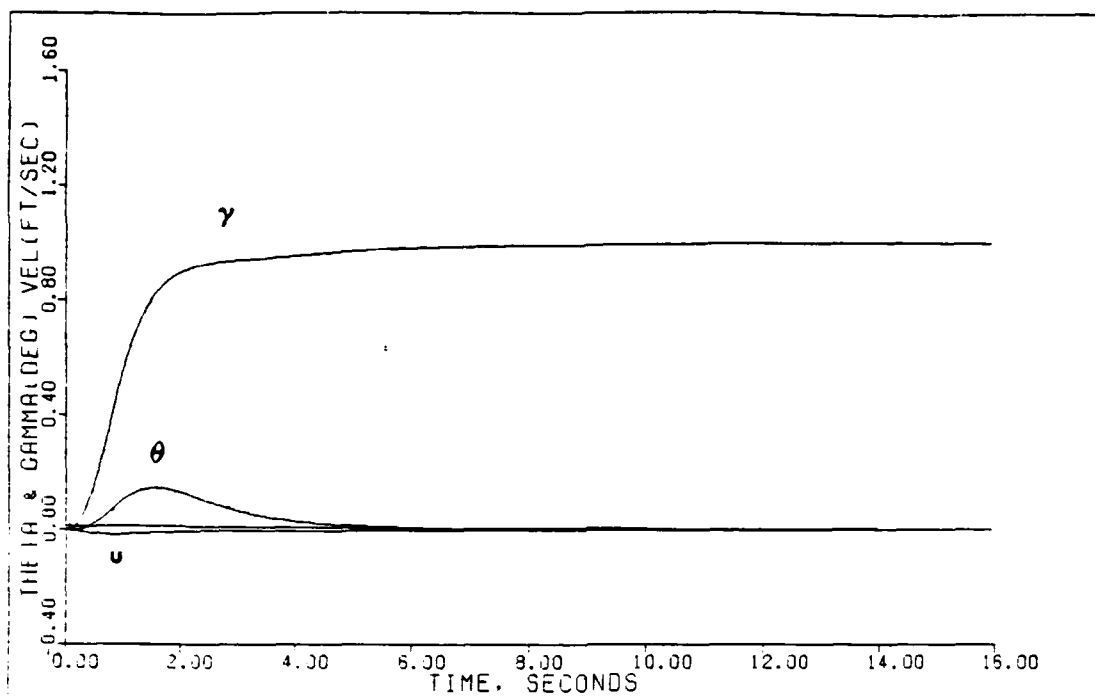
VERTICAL TRANS: PLANT-ACTUATORS-DELAY-SENSORS (1.4M FLOOD)



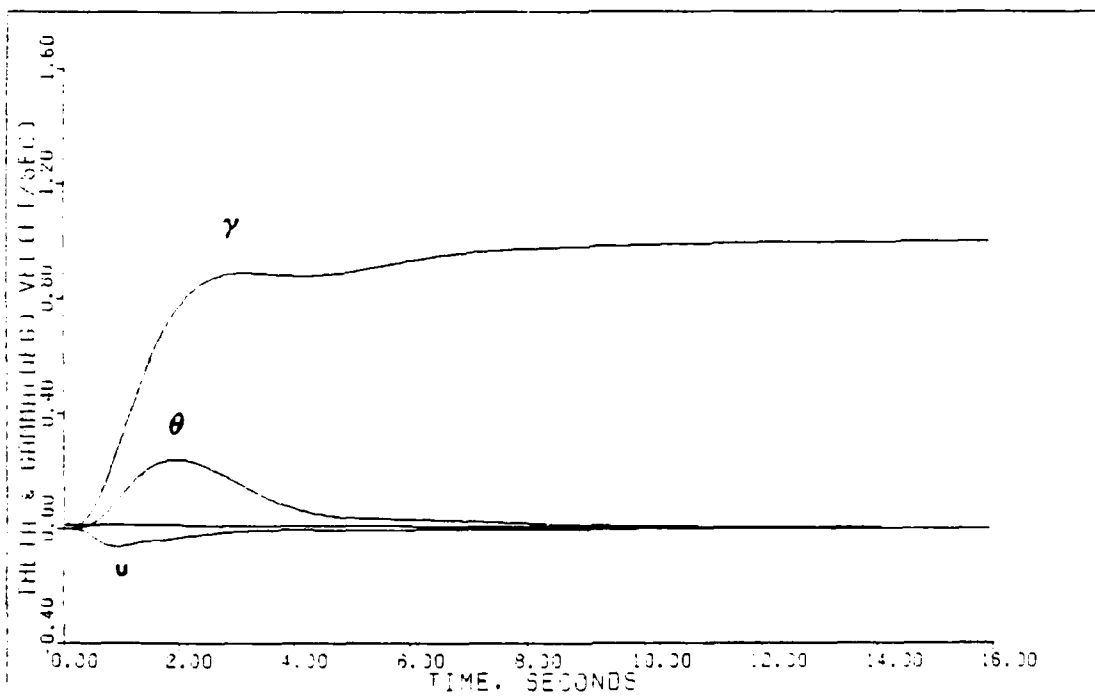
VERTICAL TRANS: PLANT-ACTUATORS (1.4M/FLOCC)



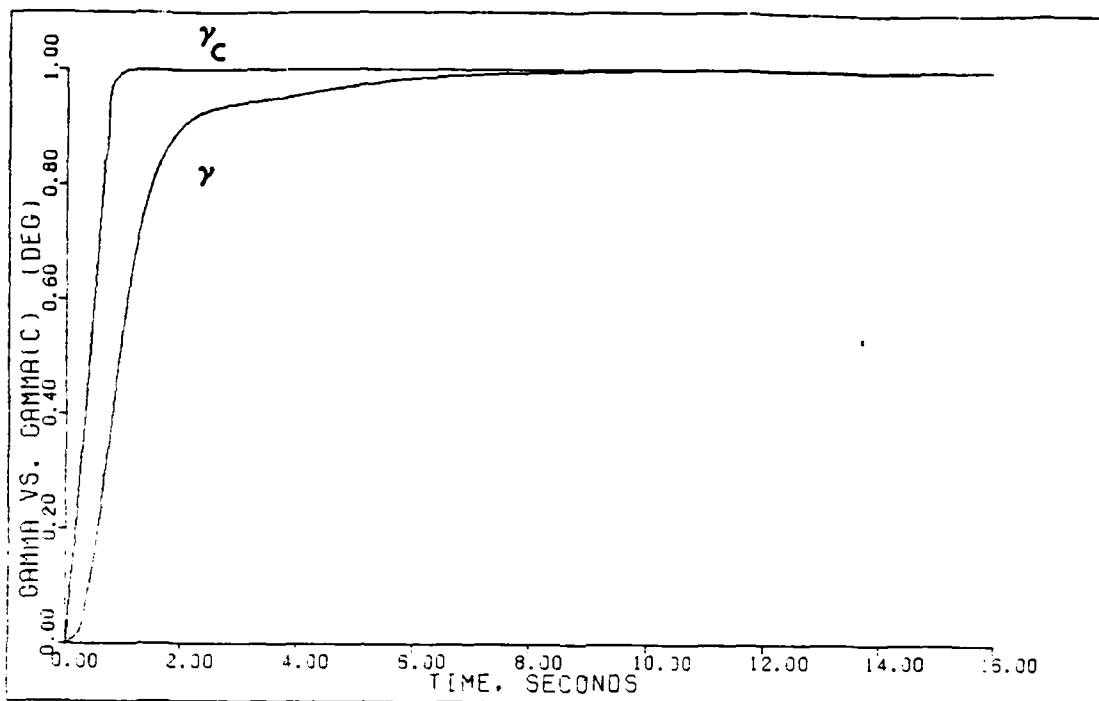
VERTICAL TRANS: PLANT-ACTUATORS+DELAY+SENSORS (1.4M/FLOCC)



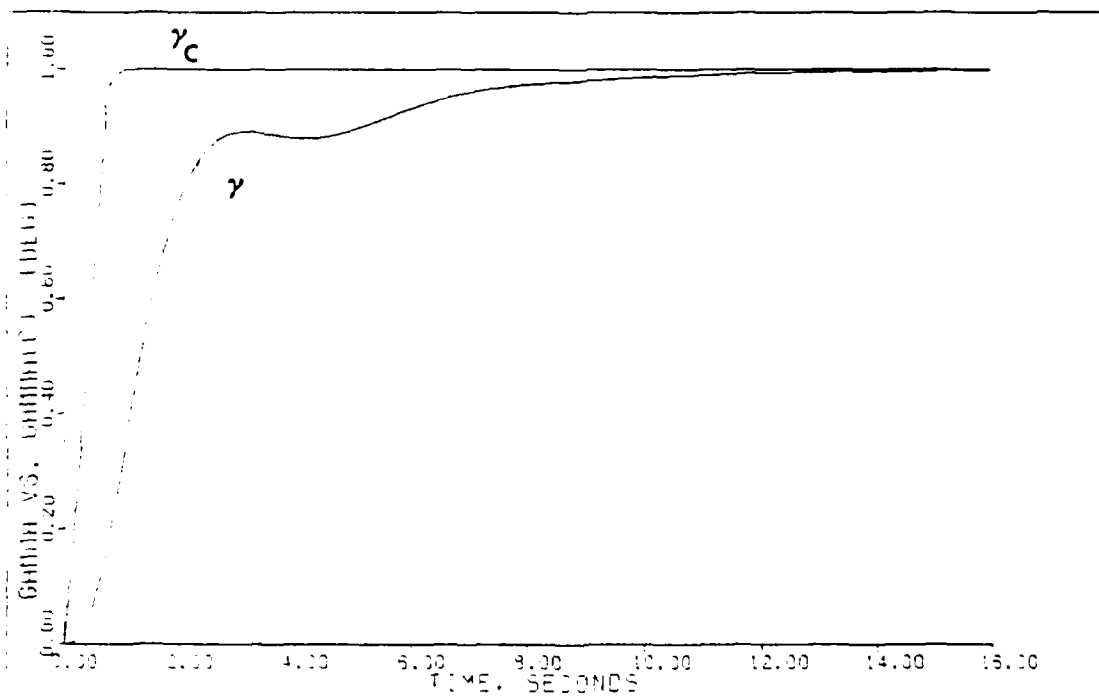
VERTICAL TRANS: PLANT+ACTUATORS (1.4M/FL200)



VERTICAL TRANS: PLANT+ACTUATORS+DELAY+SENSORS (1.4M/FL200)



VERTICAL TRANS: PLANT-ACTUATORS (1.4M/FL200)



VERTICAL TRANS: PLANT-ACTUATORS-DELAY-SENSORS (1.4M/FL200)

TABLE D.6

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Vertical Translation (+0.95 degs)

Flight Condition: 2.0 Mach at FL 400

Command Vector \underline{v} : v_1 = Theta: 0, 0, 0, 0
 v_2 = Velocity: 0, 0, 0, 0
 v_3 = Gamma: 0.8, 0.01658, 20, 20

Plant + Actuators

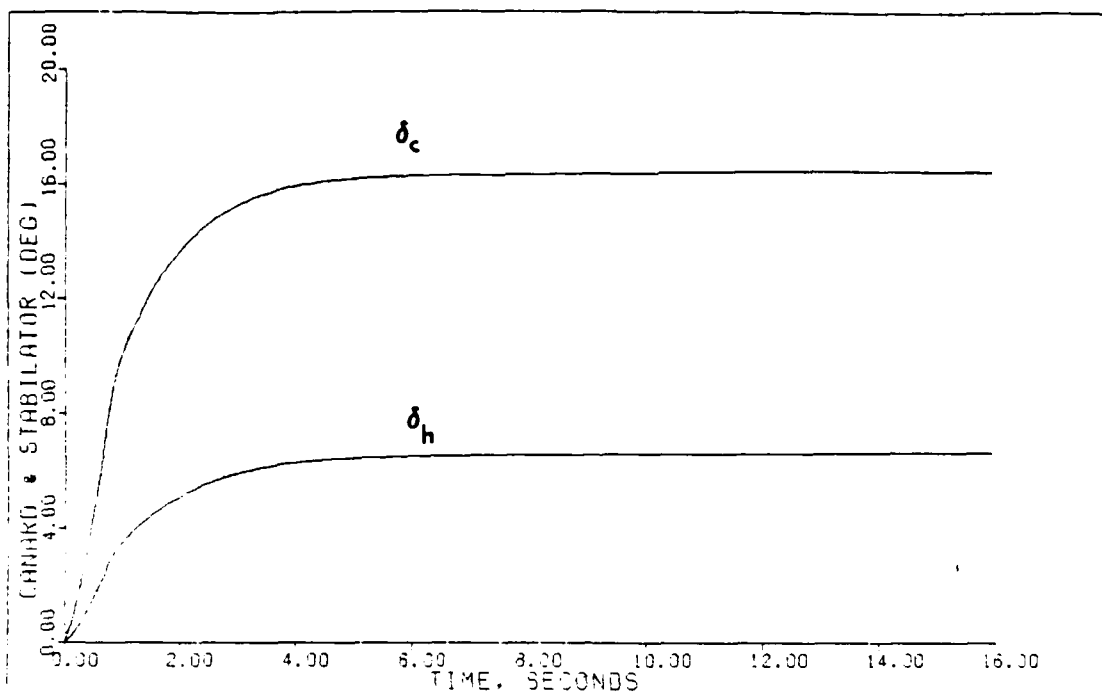
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
1.428	0.770	1.0	.1080E+02	.0000E+00	.1002E+02
		1.0	-.3335E+01	.0000E+00	.3360E+01
		0.01	-.5248E-02	.7700E-02	.1229E-01

Plant + Actuators + Delay + Sensors

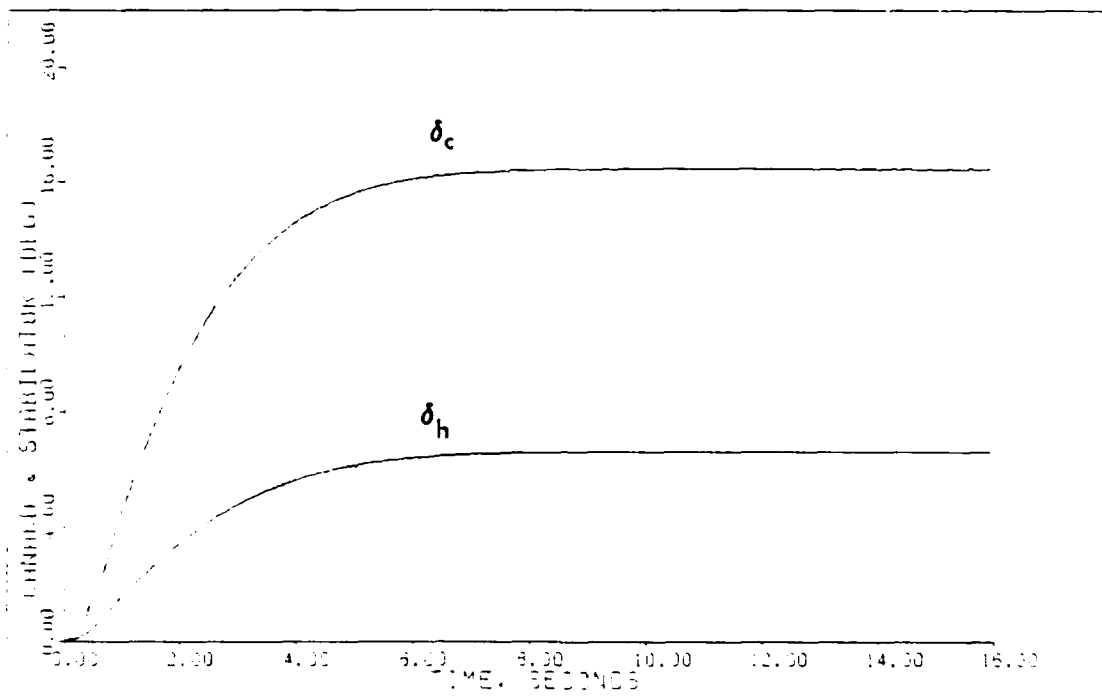
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
5.000	0.300	1.0	.4208E+01	.0000E+00	.3905E+01
		0.5	-.1299E+01	.0000E+00	.1309E+01
		0.01	-.2045E-02	.1500E-02	.4748E-02

Notes:

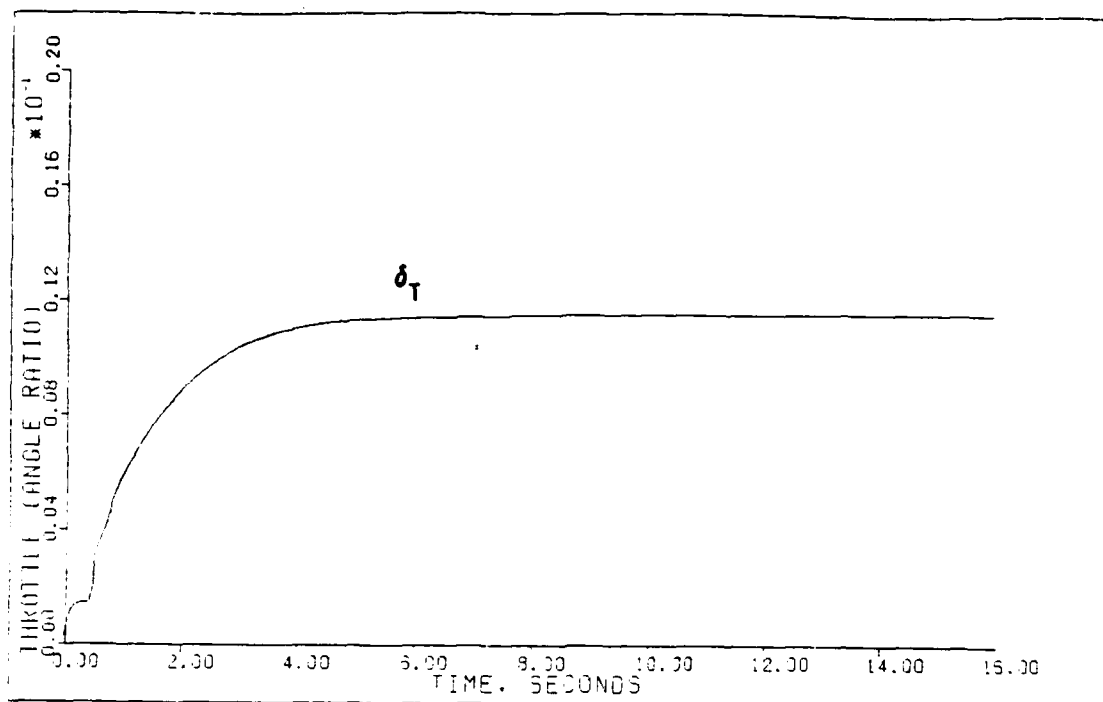
- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\alpha)\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



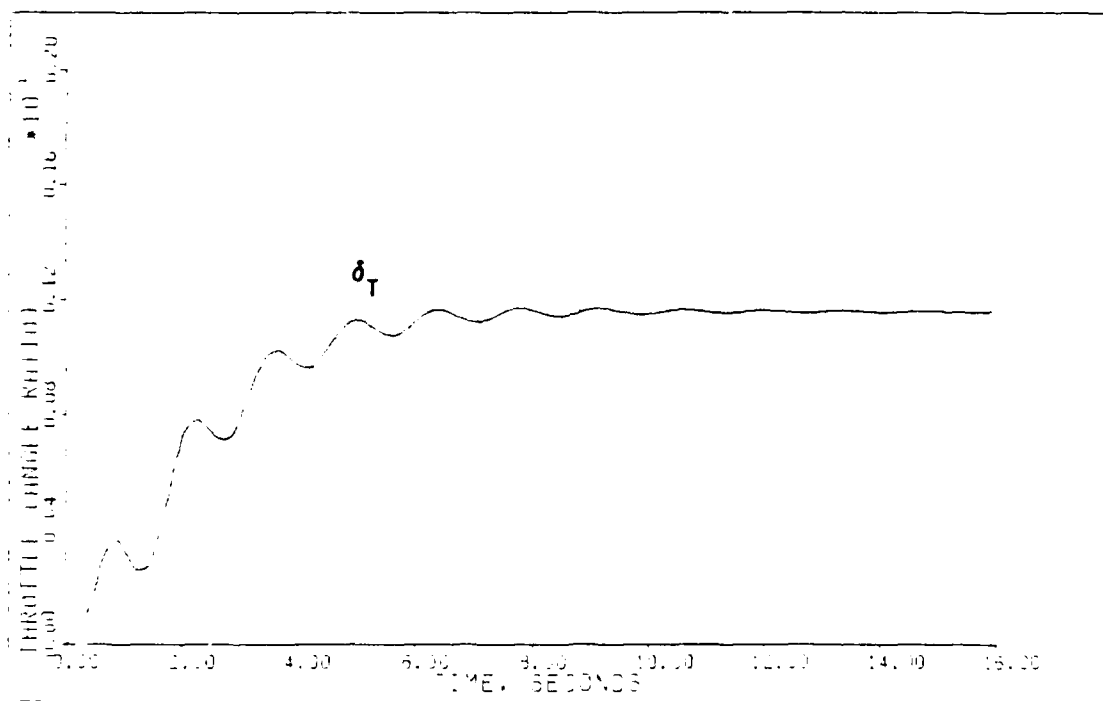
VERTICAL TRANS: PLANT-ACTUATORS (2.0M FL400)



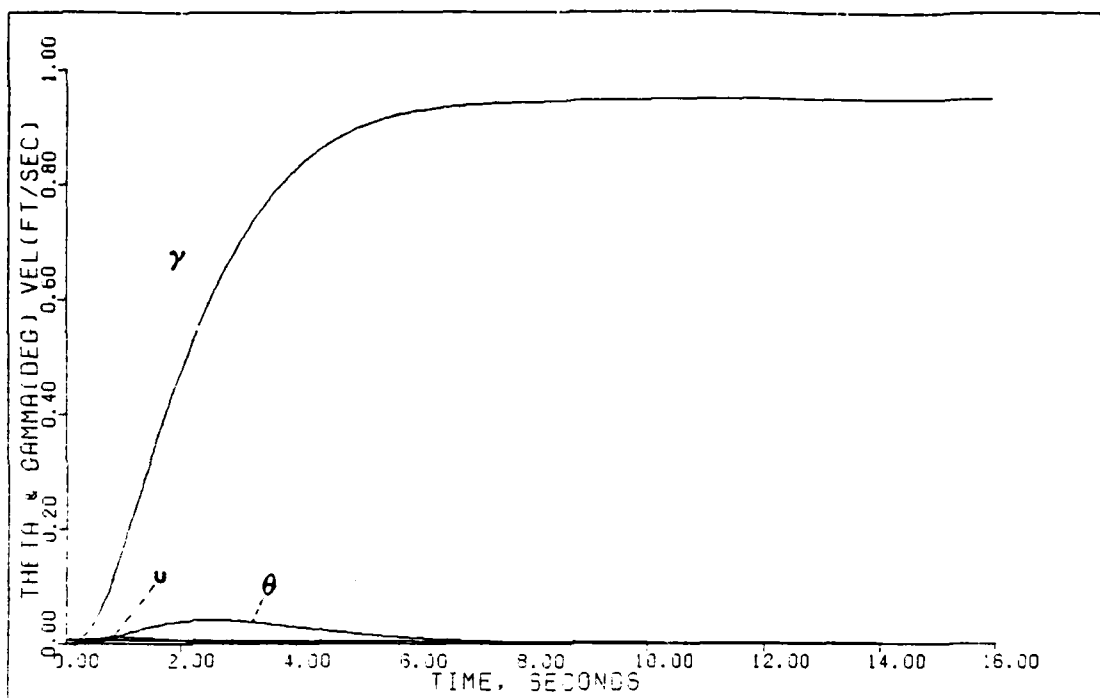
VERTICAL TRANS: PLANT-ACTUATORS-DELAY-SENSORS (2.0M FL400)



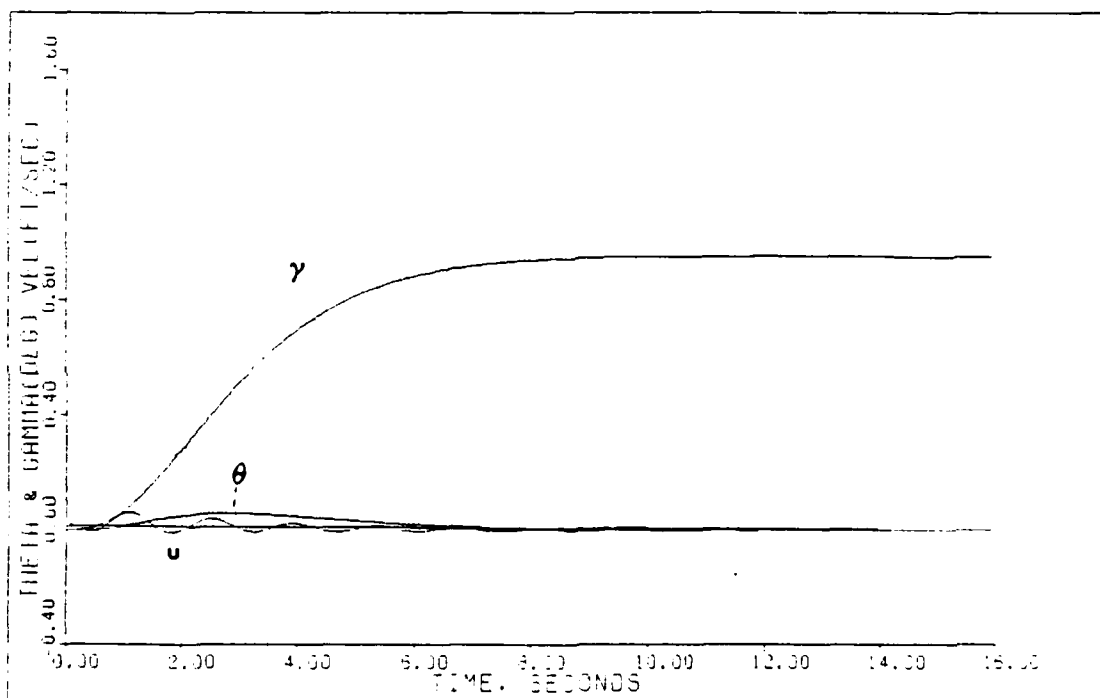
VERTICAL TRANS: PLANT+ACTUATORS (0.0M FL400)



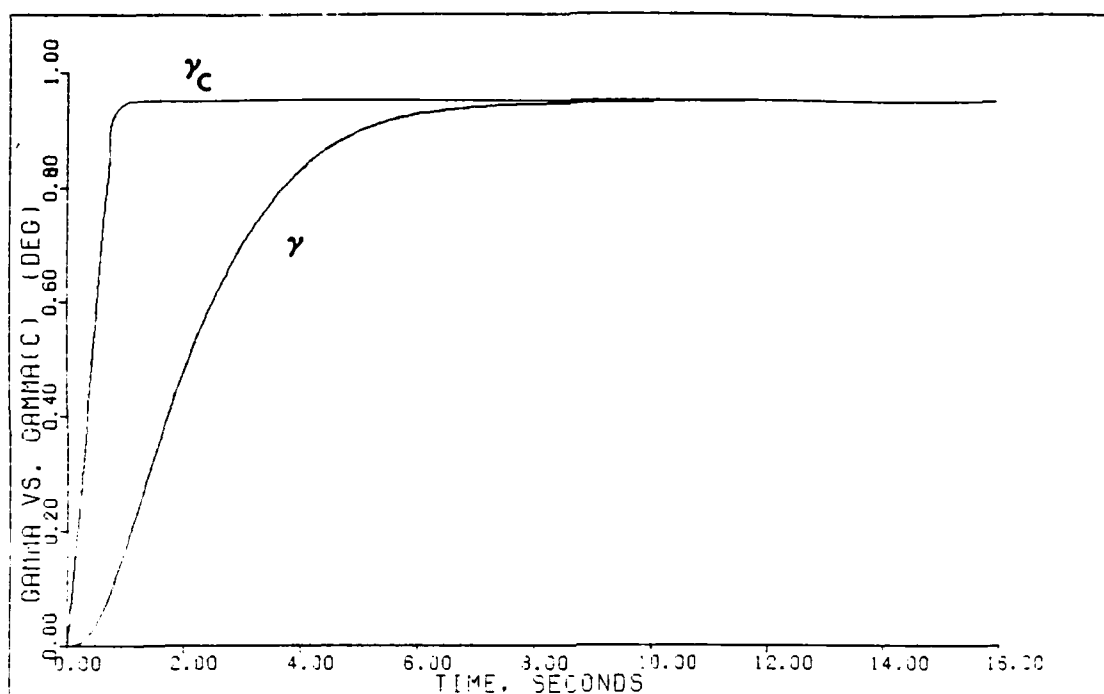
VERTICAL TRANS: PLANT+ACTUATORS+DELAY+SENSORS (0.0M FL400)



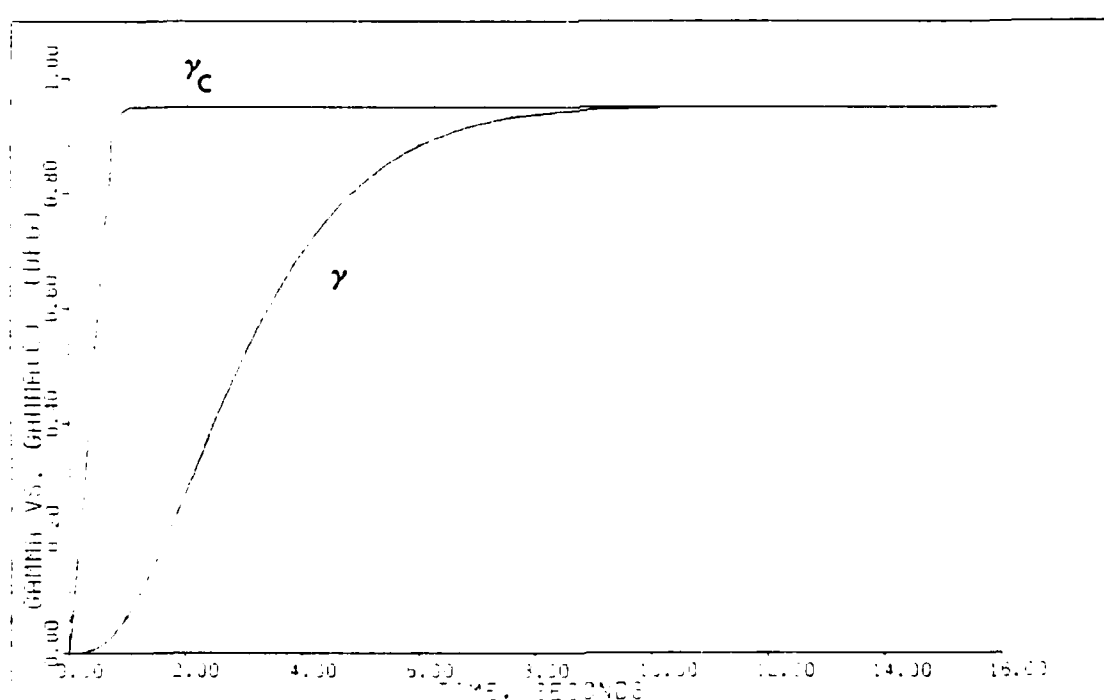
VERTICAL TRANS: PLANT-ACTUATORS (2.0M/FL400)



VERTICAL TRANS: PLANT-ACTUATORS+DELAY+SENSORS (2.0M/FL400)



VERTICAL TRANS: PLANT+ACTUATORS (2.0M/FL400)



VERTICAL TRANS: PLANT+ACTUATORS+DELAY+SENSORS (2.0M/FL400)

TABLE D.7
DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Pitch Pointing (+2.5 degs)

Flight Condition: 0.3 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 1.2, 0.04363, 20, 20$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0, 0, 0, 0$

Plant + Actuators

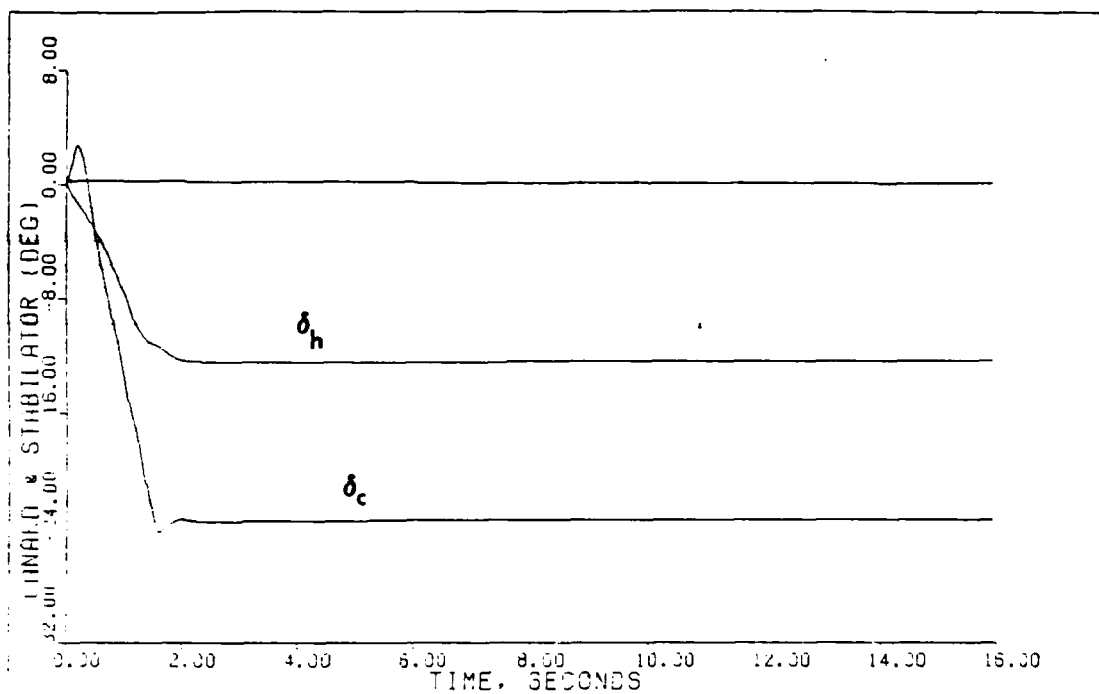
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>K₀</u>		
1.111	0.720	1.2	.9230E+02	.0000E+00	.8340E+03
		1.5	-.2601E+02	.0000E+00	.4727E+03
		1.1	.6135E-01	.5400E-01	.2202E+01

Plant + Actuators + Delay + Sensors

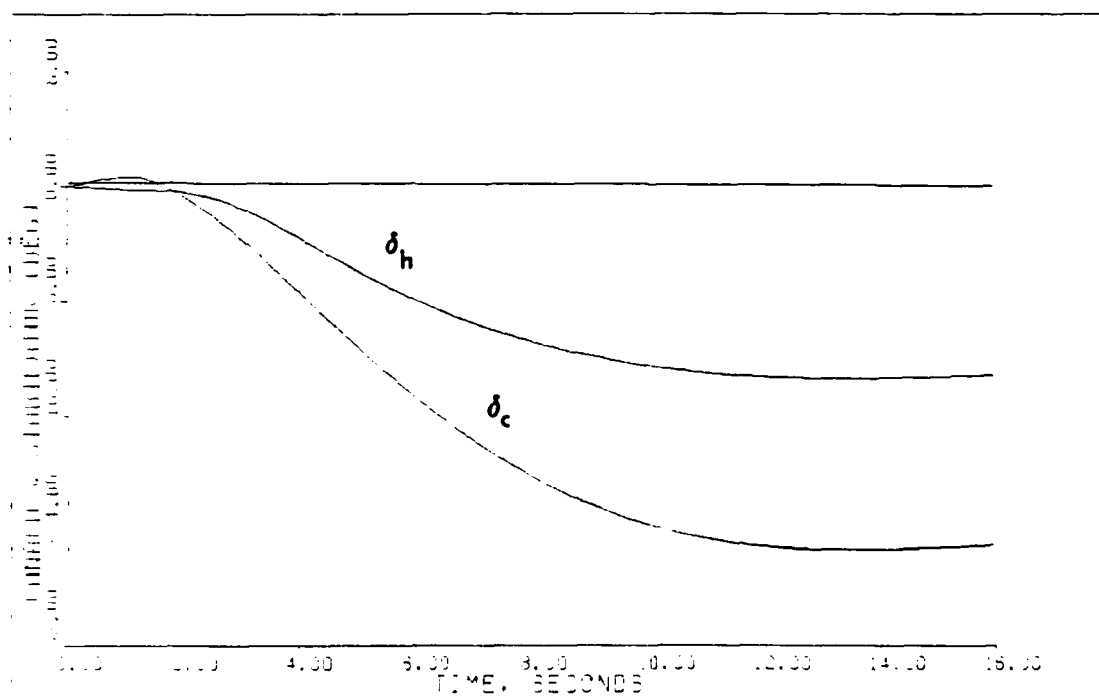
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>K₀</u>		
1.428	0.070	1.2	.1282E+01	.0000E+00	.5897E+01
		1.5	-.3613E+00	.0000E+00	.3342E+01
		0.08	.8521E-03	.5250E-02	.1557E-01

Notes:

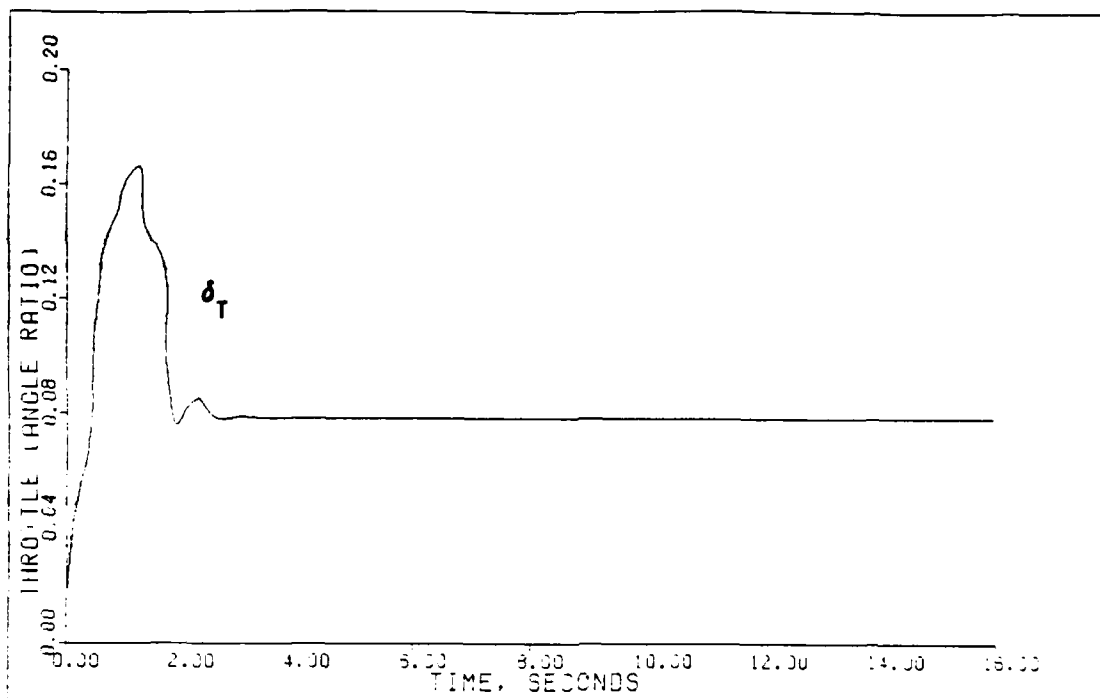
1. Each \underline{v} input is composed of four parts:
 - A. Time (secs) that the input reaches steady-state.
 - B. Steady-state value (radians).
 - C. Time (secs) input leaves steady-state.
 - D. Time (secs) input reaches zero.
2. Sigma = the elements (in order) of the diagonal matrix.
3. The integral controller matrix $\underline{K}_1 = (\alpha)\underline{K}_0$.
4. Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$ (Plant + Actuators).
 $\underline{\bar{M}} = \{2.1, 0, 0\}^T$ (Plant + Act + TD + Sen).



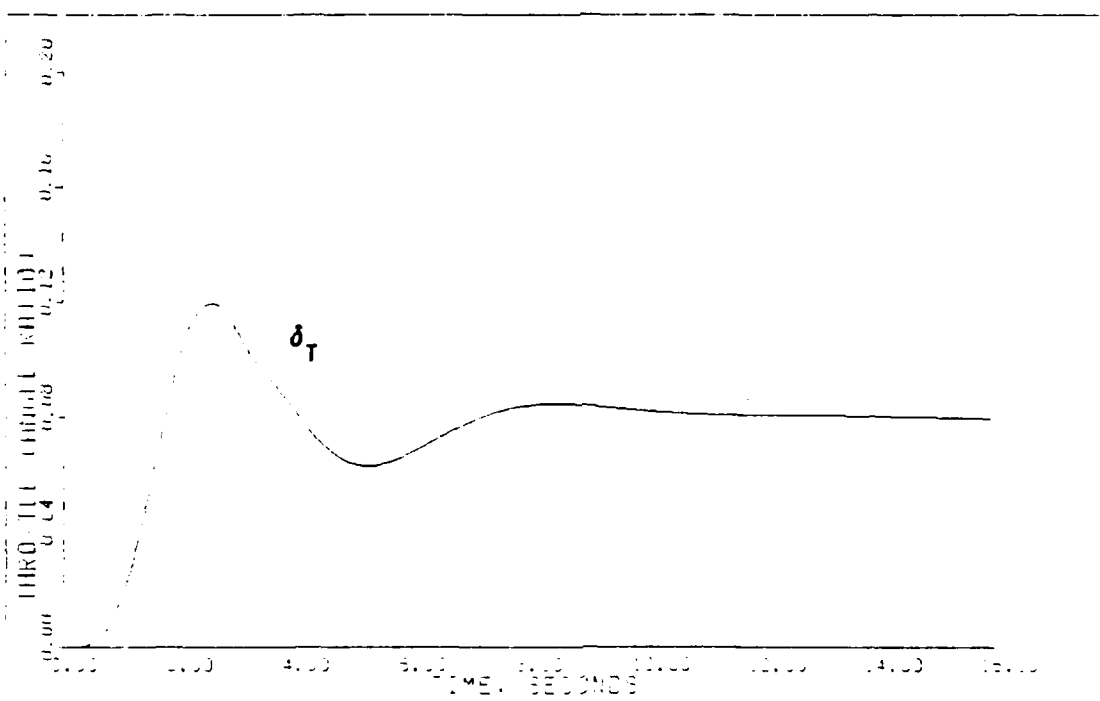
PITCH POINTING: PLANT-ACTUATORS (0.3M/FLECC)



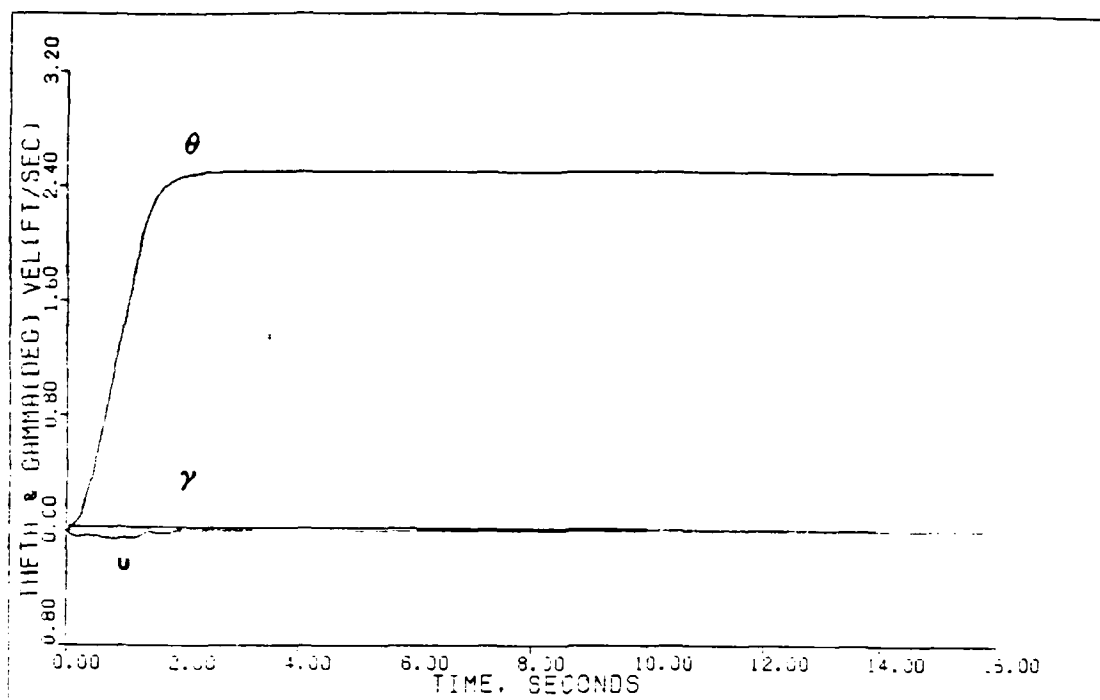
PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (0.3M/FLECC)



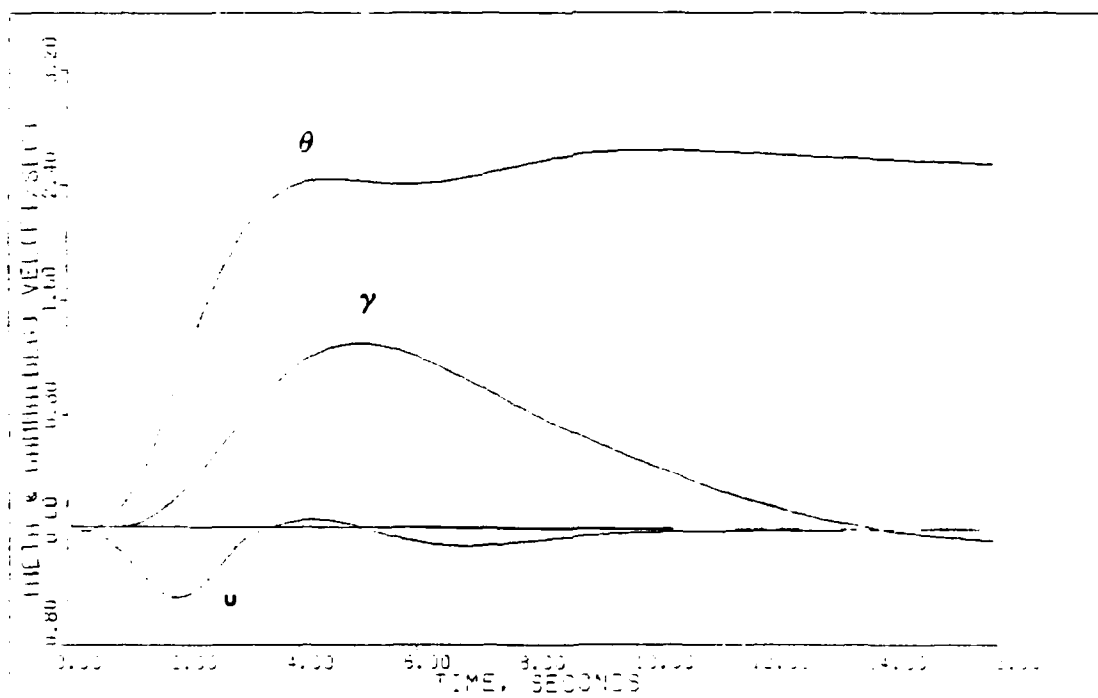
PITCH POINTING: PLANT-ACTUATORS (0.8M/FLESC)



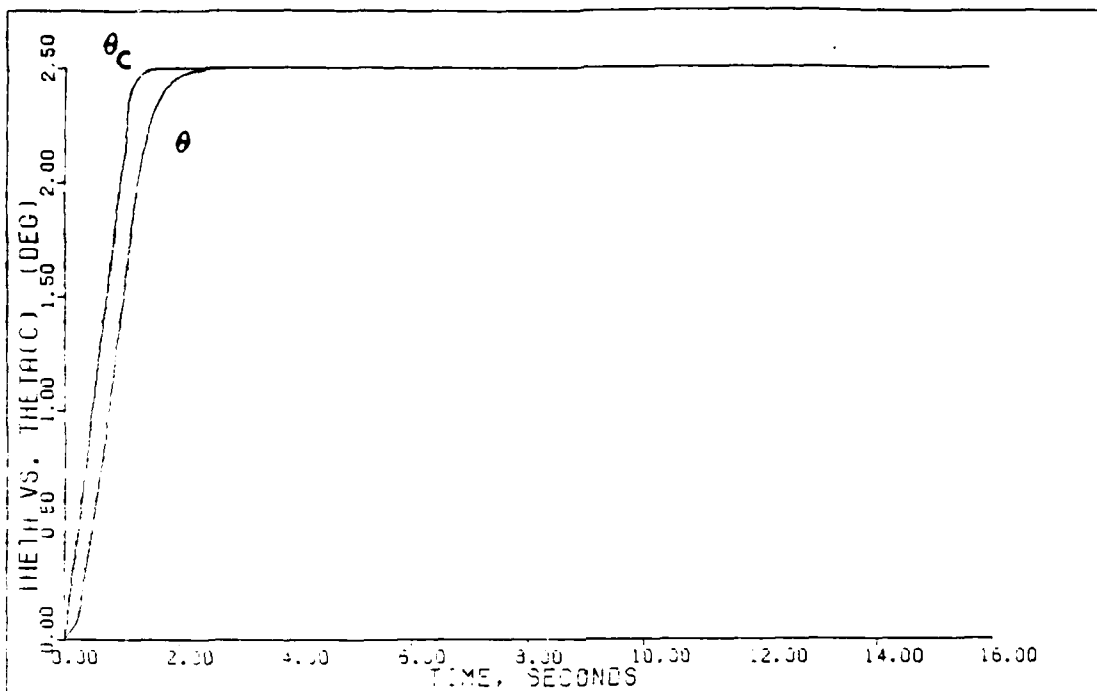
PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (0.8M/FLESC)



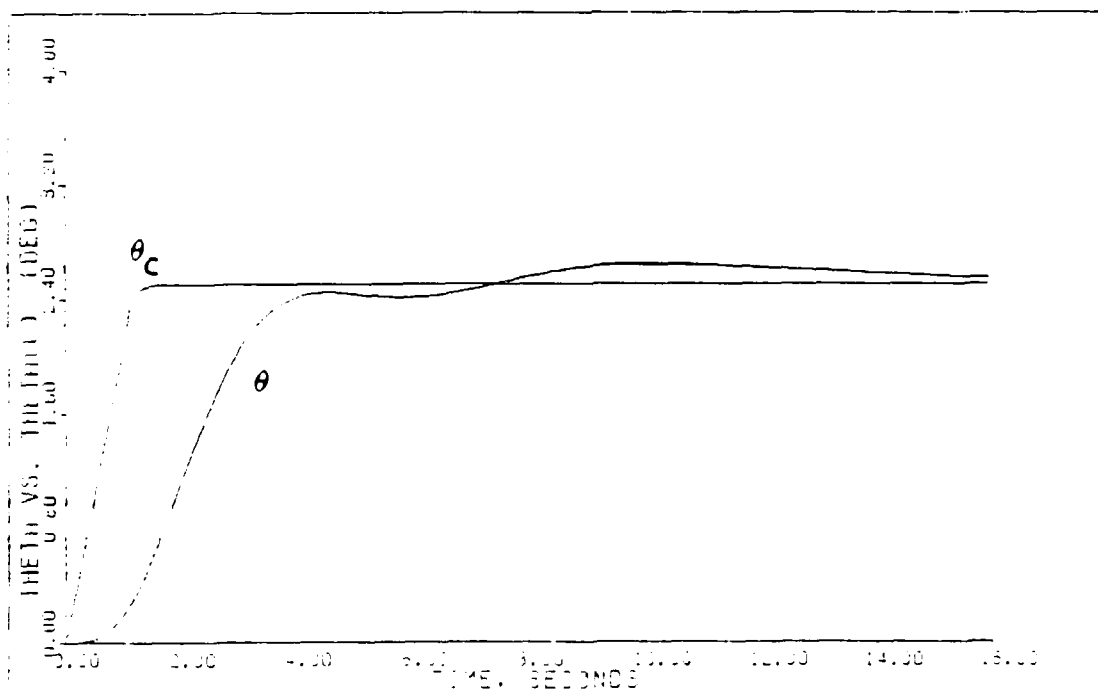
PITCH POINTING: PLANT-ACTUATORS (0.3M/FL200)



PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (0.3M/FL200)



PITCH POINTING: PLANT-ACTUATORS (0.3M/FL200)



PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (0.3M/FL200)

TABLE D.8

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Pitch Pointing (+2.9 degs)

Flight Condition: 0.9 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 0.8, 0.05061, 20, 20$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0, 0, 0, 0$

Plant + Actuators

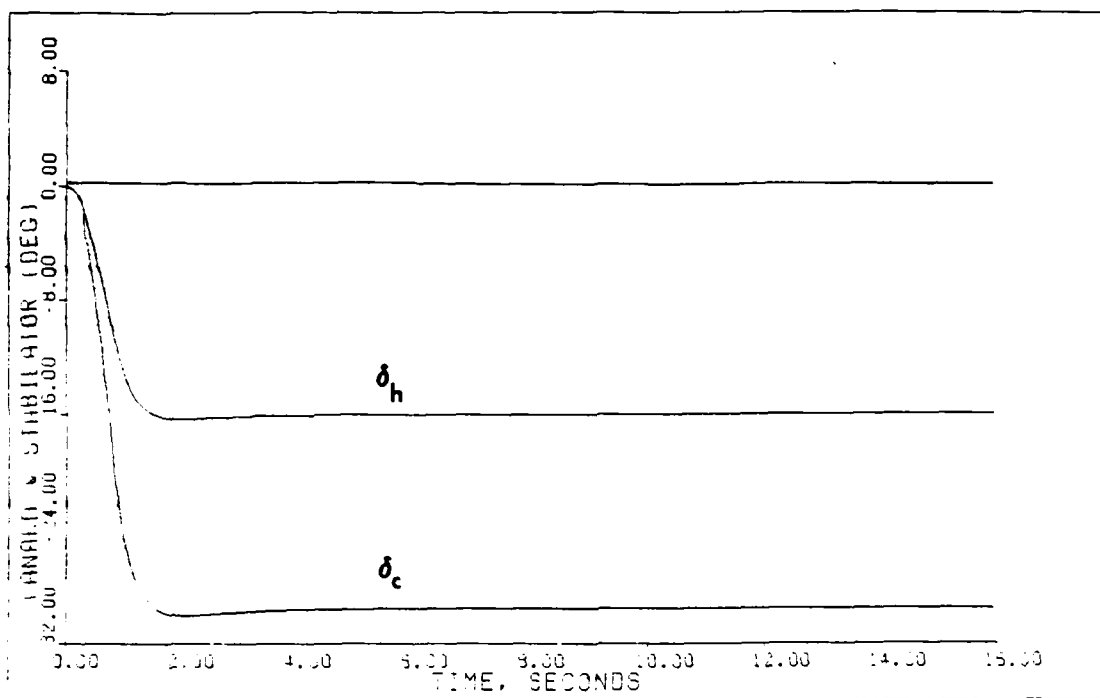
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
1.111	0.999	1.2	.1389E+02	.0000E+00	.4009E+03
		0.8	-.3351E+01	.0000E+00	.2385E+03
		1.1	-.2066E-01	.1776E-01	.7007E+00

Plant + Actuators + Delay + Sensors

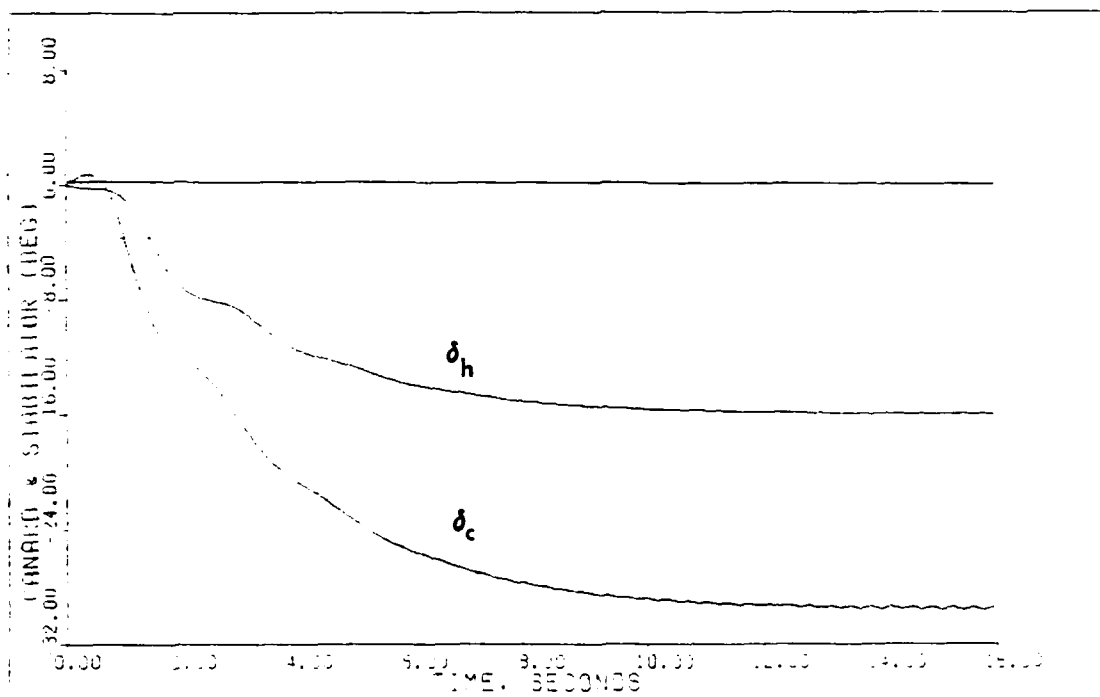
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
1.250	0.240	1.2	.3337E+01	.0000E+00	.7004E+01
		0.8	-.8050E+00	.0000E+00	.4168E+01
		0.08	-.4964E-02	.4267E-02	.1224E-01

Notes:

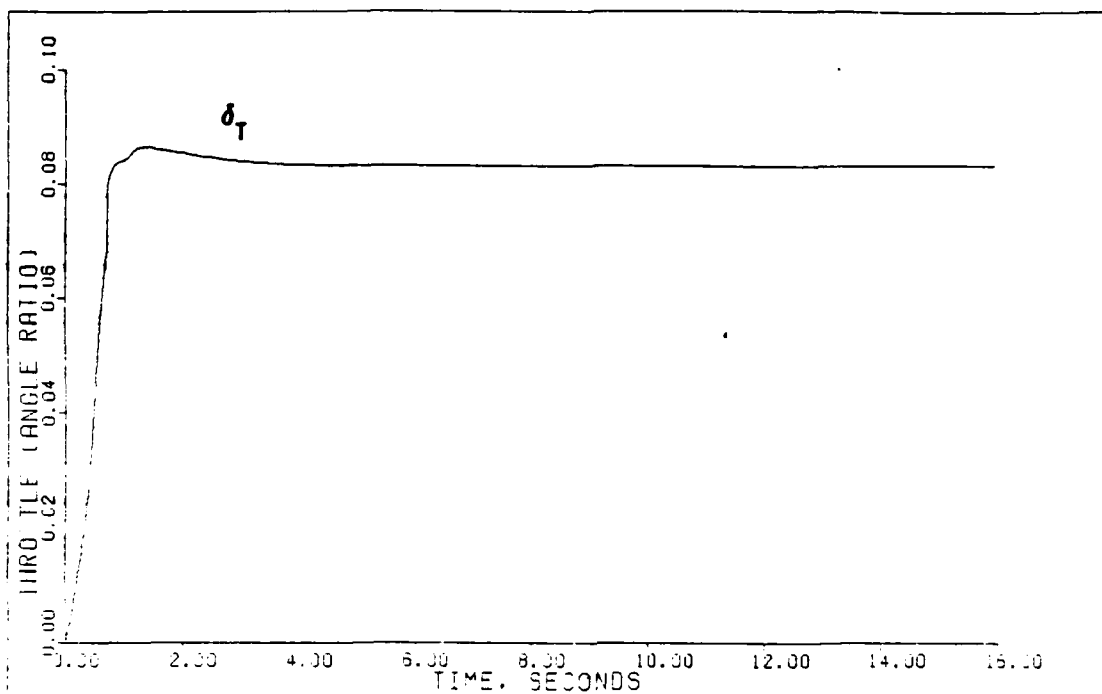
- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\text{alpha})\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



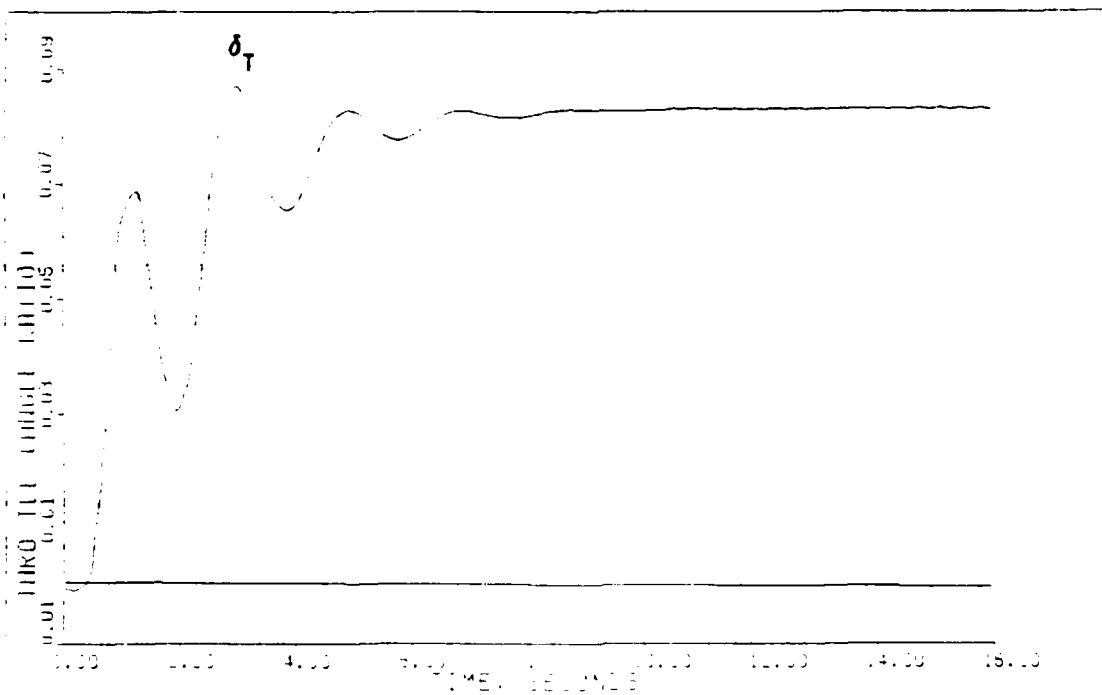
PITCH POINTING: PLANT-ACTUATORS (0.8M/F1200)



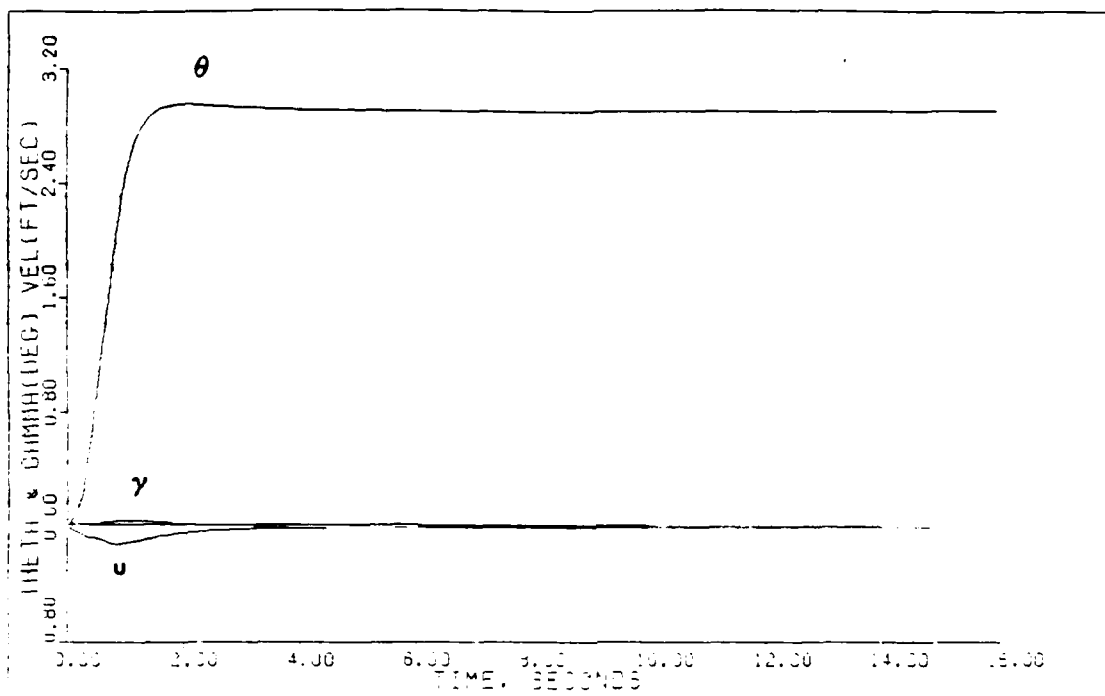
PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (0.8M/F1200)



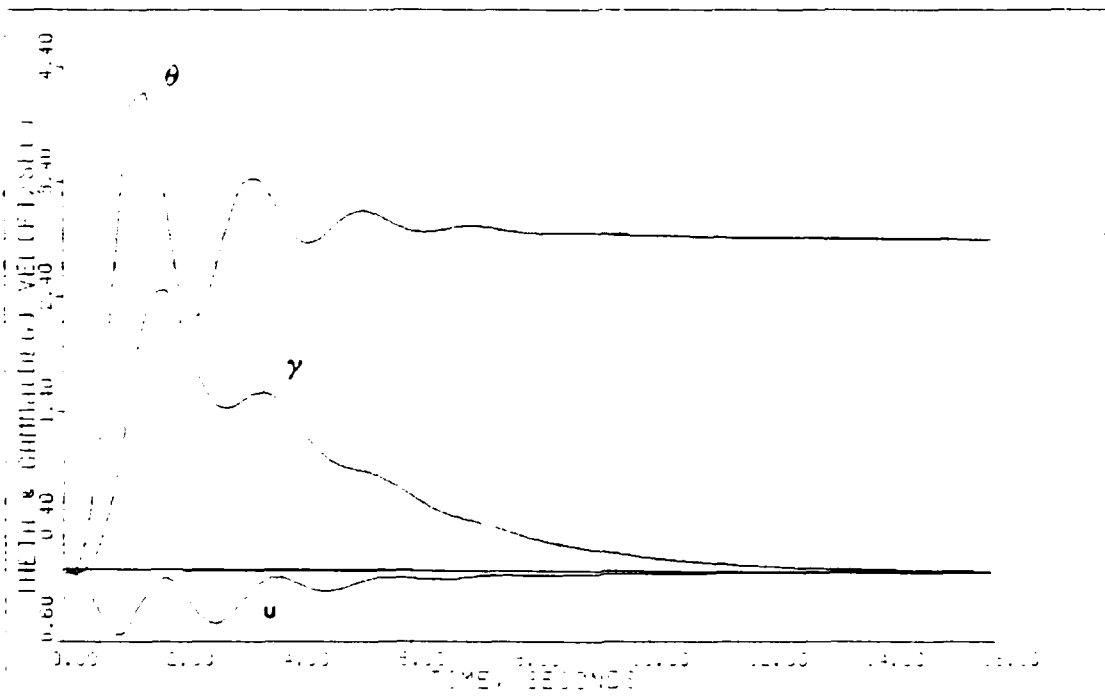
PITCH POINTING: PLANT-ACTUATORS (0.9M/FL200)



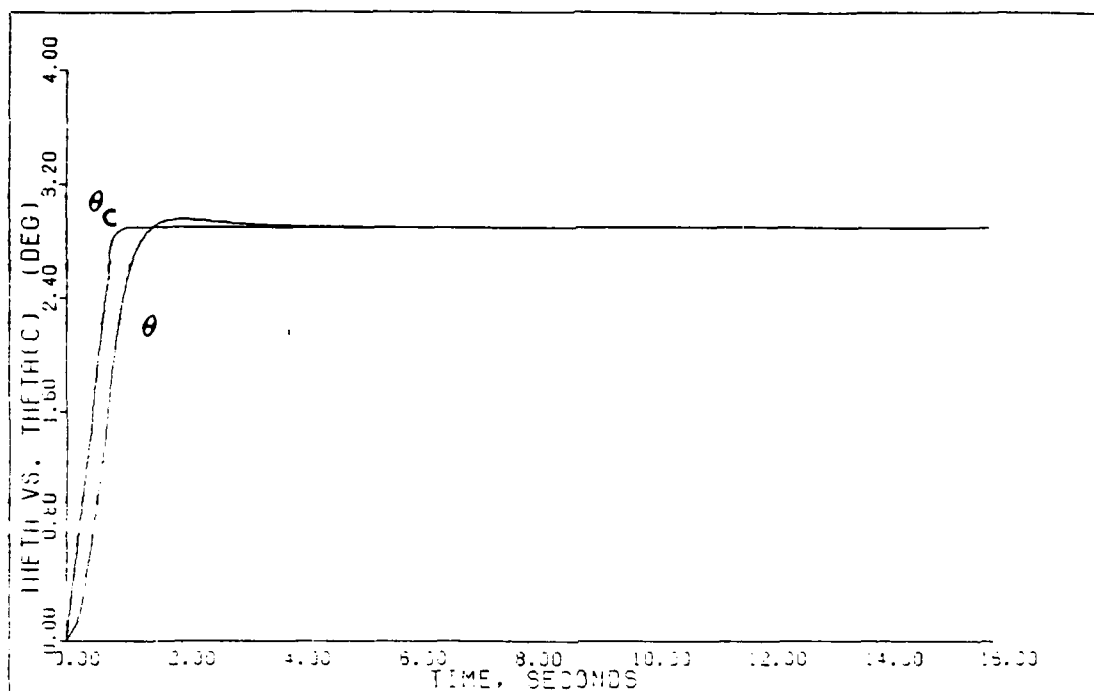
PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (0.9M/FL200)



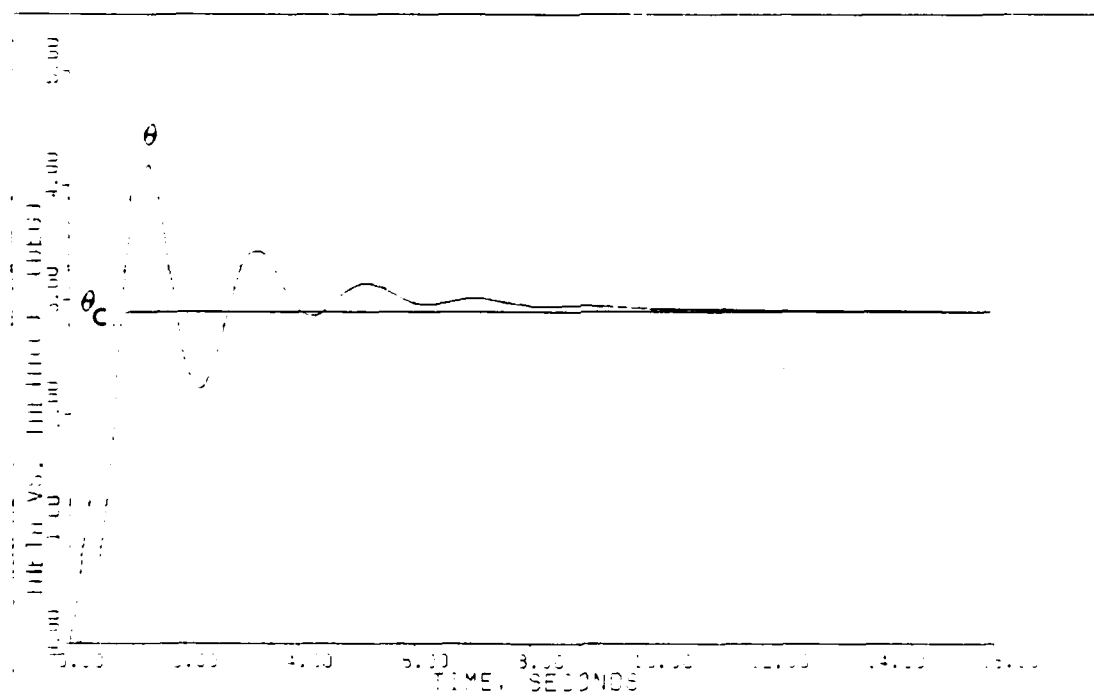
PITCH POINTING: PLANT-ACTUATOR (0.9M, FL200)



PITCH POINTING: PLANT-ACTUATOR-OSLA-SENSORS (0.9M, FL200)



PITCH POINTING: PLANT-ACTUATORS (0.9M, FL200)



PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (0.9M, FL200)

TABLE D.9
DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Pitch Pointing (+1.9 degs)

Flight Condition: 2.0 Mach at FL 400

Command Vector \underline{v} : $v_1 = \text{Theta: } 0.8, 0.03316, 20, 20$
 $v_2 = \text{Velocity: } 0, 0, 0, 0$
 $v_3 = \text{Gamma: } 0, 0, 0, 0$

Plant + Actuators

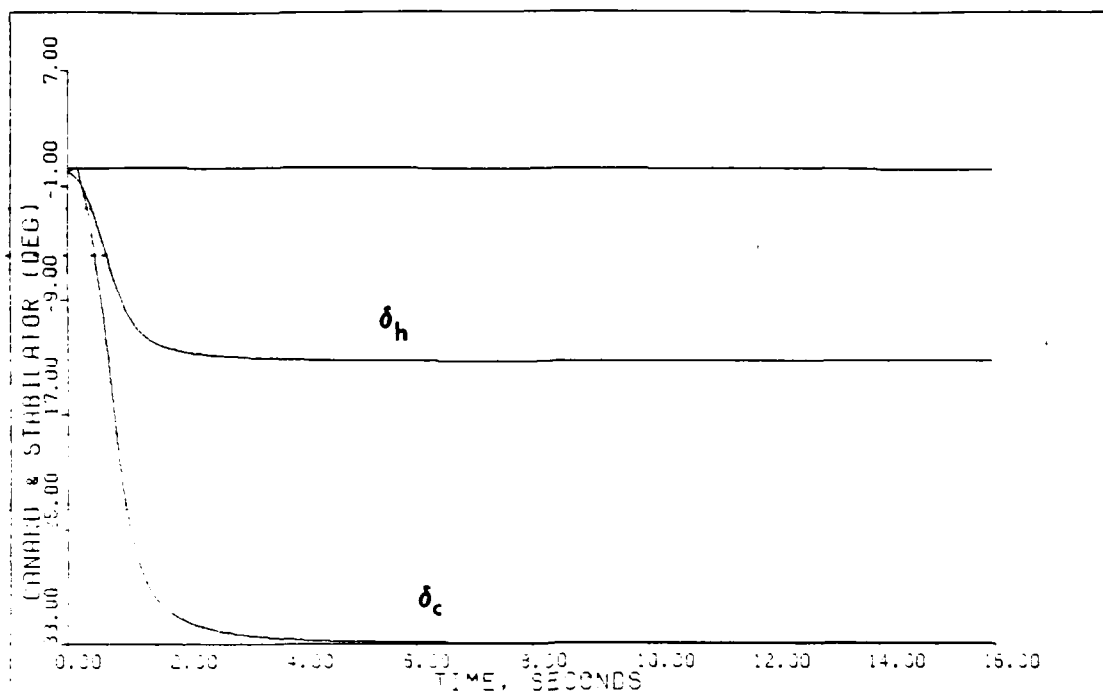
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
1.111	0.999	1.05	.1471E+02	.0000E+00	.1300E+04
		0.9	-.4543E+01	.0000E+00	.4359E+03
		1.0	-.7149E-02	.8991E-02	.1594E+01

Plant + Actuators + Delay + Sensors

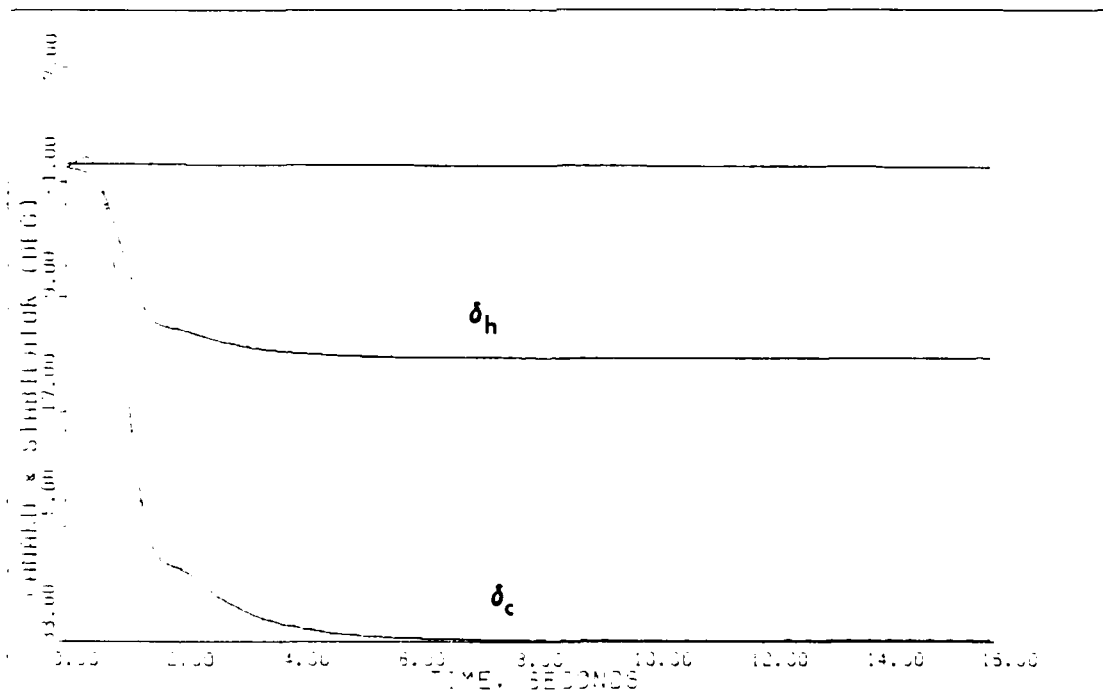
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>		
2.0	0.270	1.05	.3976E+01	.0000E+00	.3514E+03
		0.9	-.1228E+01	.0000E+00	.1178E+03
		1.0	-.1932E-02	.2430E-02	.4308E+00

Notes:

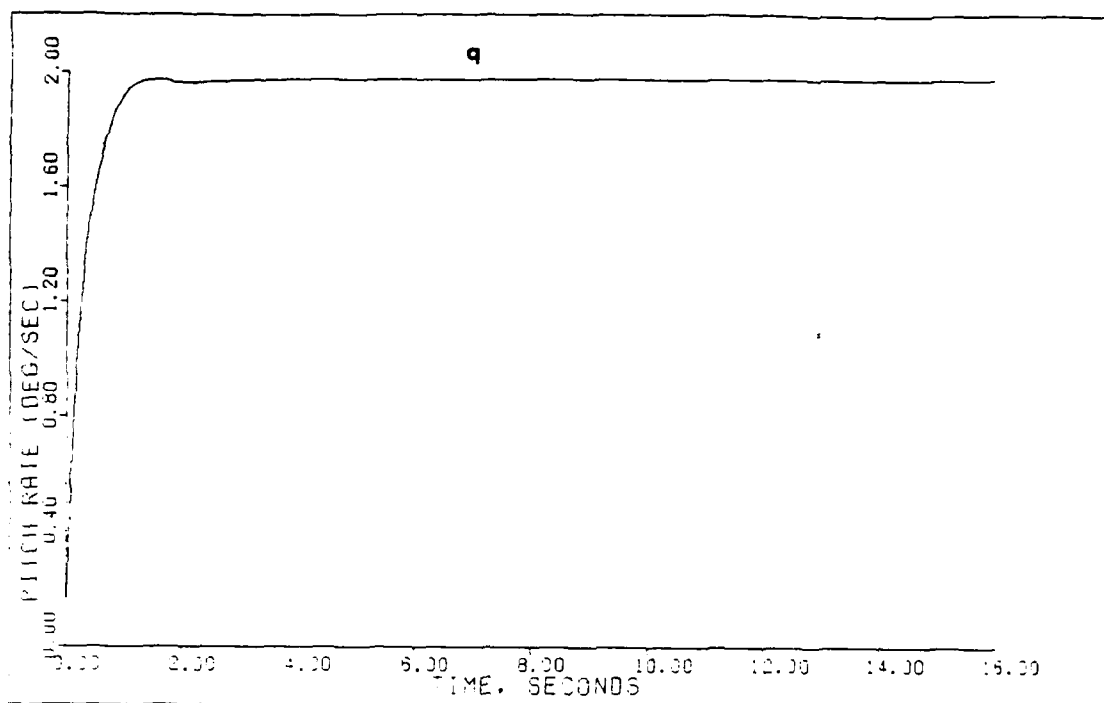
1. Each \underline{v} input is composed of four parts:
 - A. Time (secs) that the input reaches steady-state.
 - B. Steady-state value (radians).
 - C. Time (secs) input leaves steady-state.
 - D. Time (secs) input reaches zero.
2. Sigma = the elements (in order) of the diagonal matrix.
3. The integral controller matrix $\underline{K}_1 = (\text{alpha})\underline{K}_0$.
4. Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



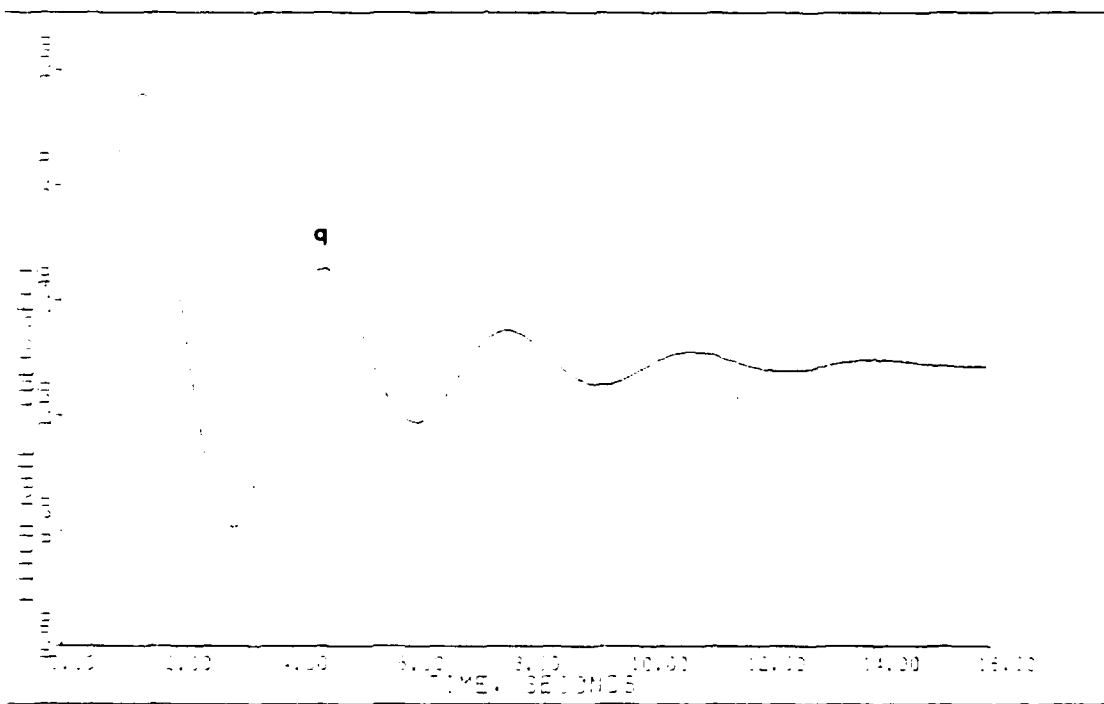
PITCH POINTING: PLANT+ACTUATORS (2.0MFL400)



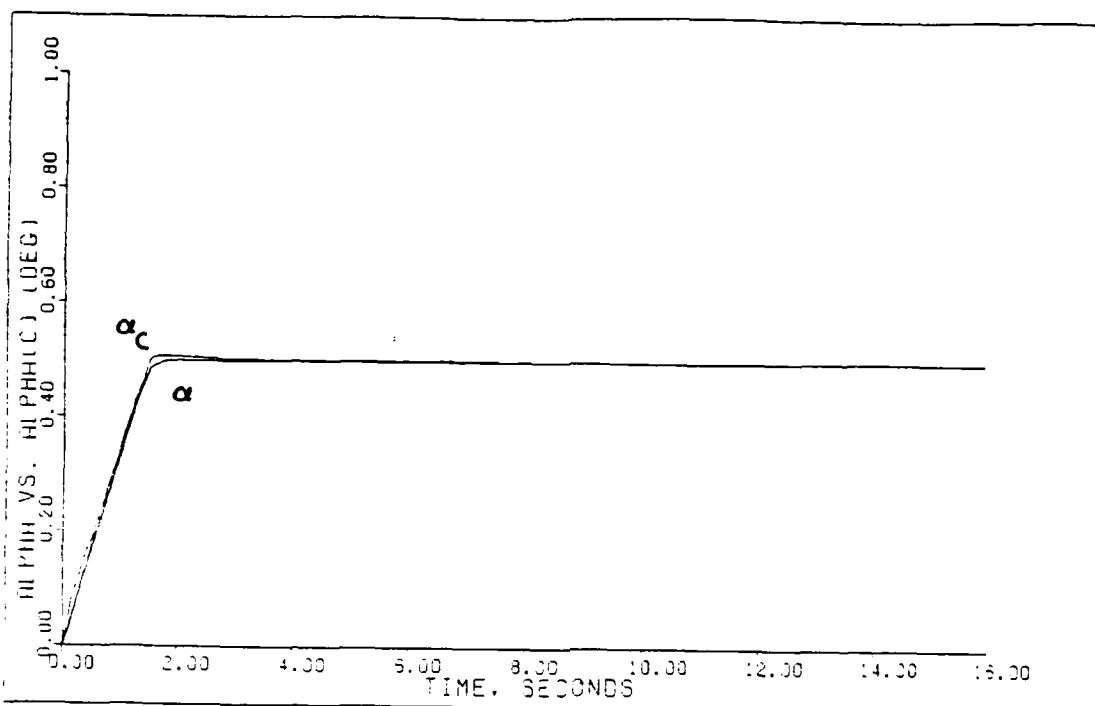
PITCH POINTING: PLANT+ACTUATORS+DELAY+SENSORS (2.0MFL400)



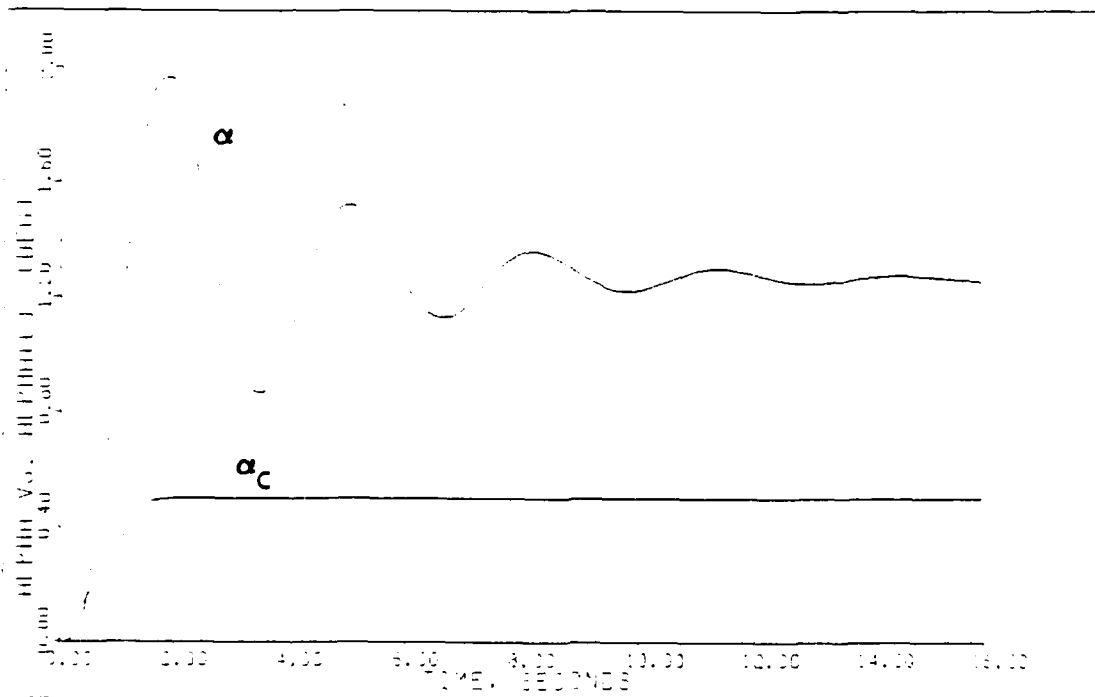
13 PULL-UP: PLANT+ACTUATORS (0.9M/FL200)



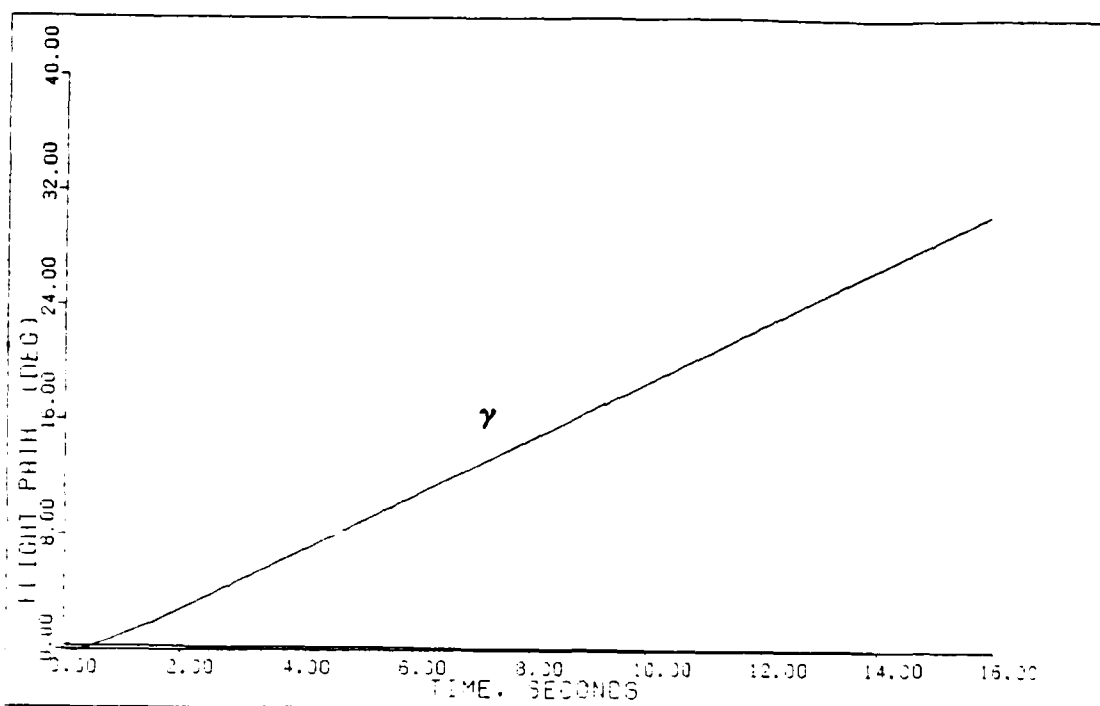
13 PULL-UP: PLANT+ACTUATORS+DELAY+Sensors (0.9M/FL200)



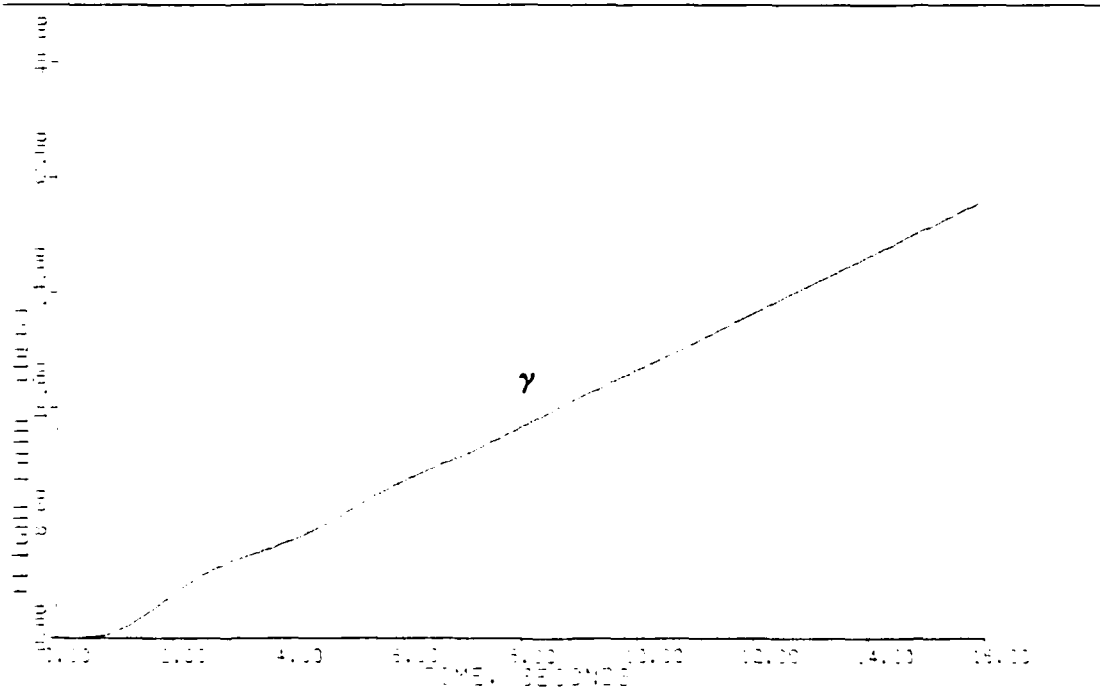
29 PULL-UP: PLANT-ACTUATORS (0.9M/FL200)



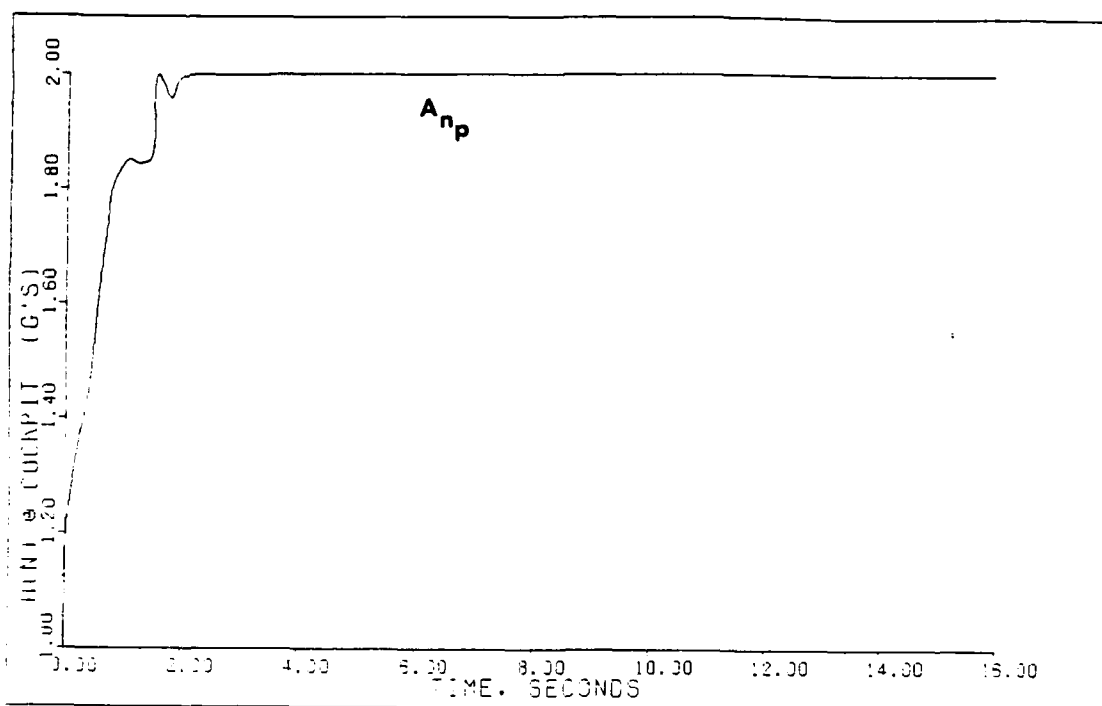
29 PULL-UP: PLANT-ACTUATORS-DELAY-SENSORS (0.9M/FL200)



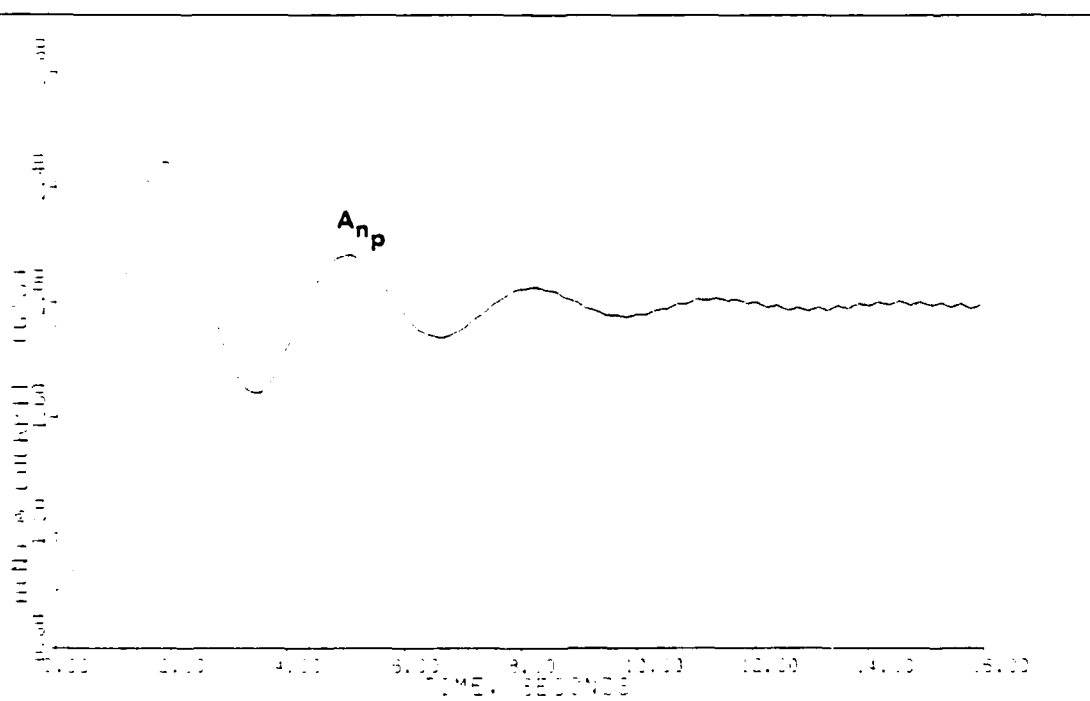
0G PULL-UP: PLANT-ACTUATORS (0.9M, FL200)



1G PULL-UP: PLANT-ACTUATORS-DELA-SENSOR (0.9M, FL200)



10 PULL-UP: PLANT-ACTUATORS (0.9M/FL200)



10 PULL-UP: PLANT-ACTUATORS-DELAY-SENSORS (0.9M FL200)

TABLE D.11

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Constant g Pull-up (9.0 g's)

Flight Condition: 0.9 Mach at FL 200

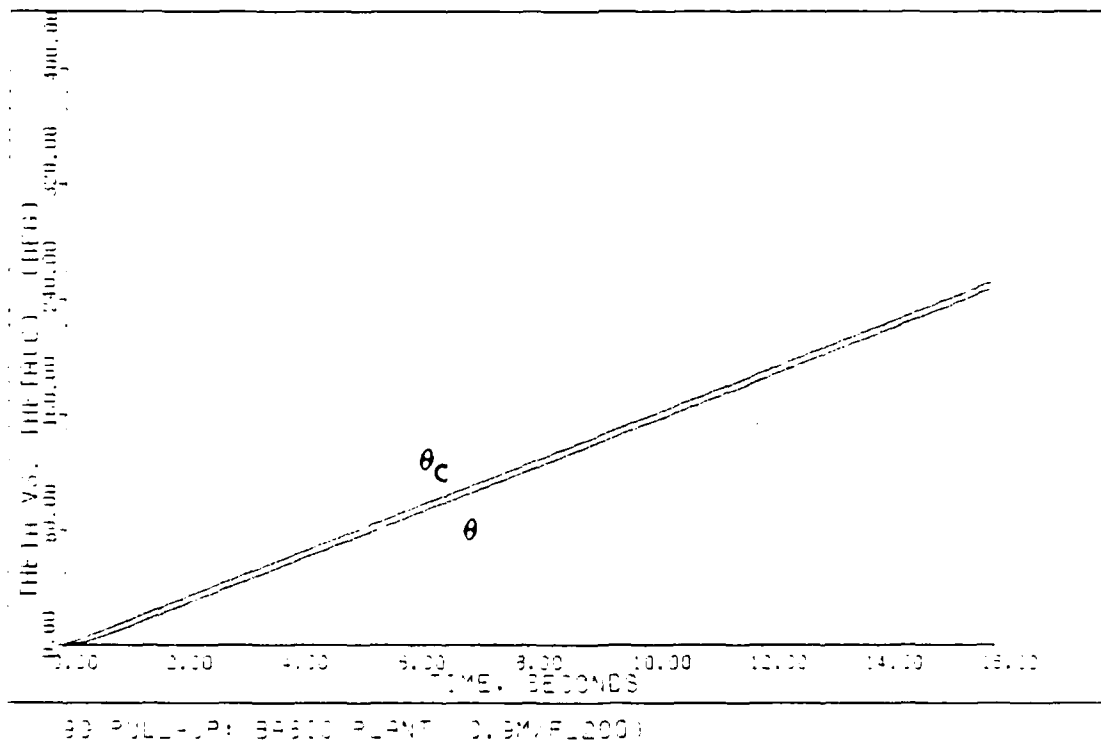
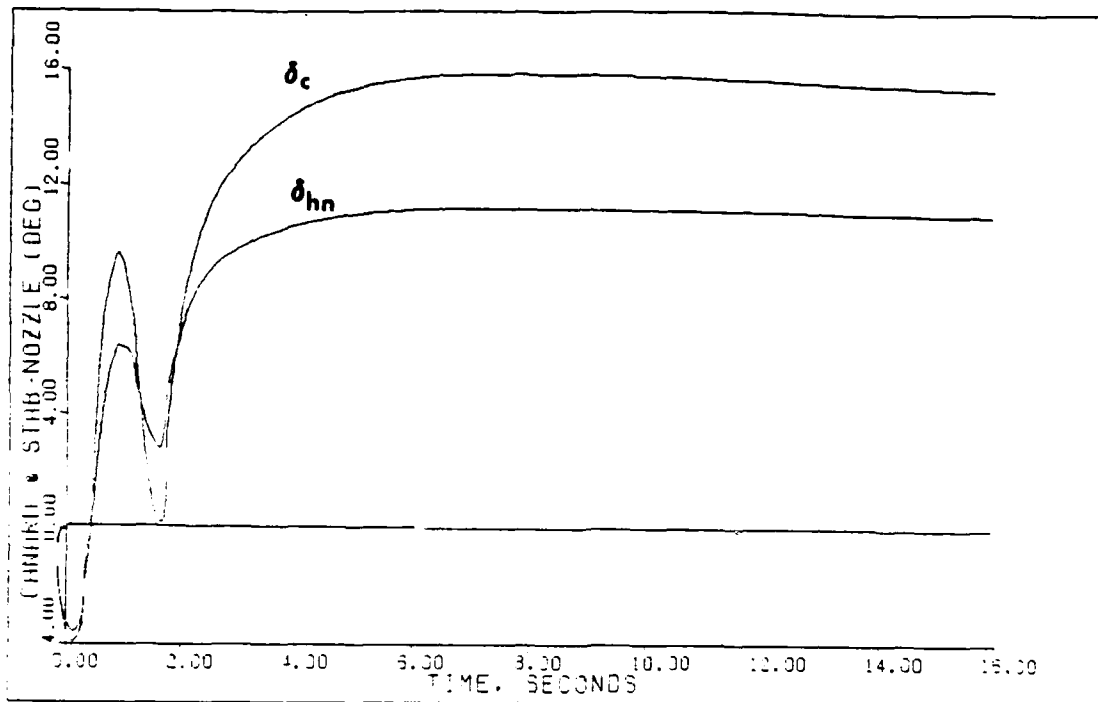
Command Vector \underline{v} : $v_1 = \text{Theta: } 20, 5.51906, 20, 20$
 $v_2 = \text{Alpha: } 1.5, 0.148353, 20, 20$

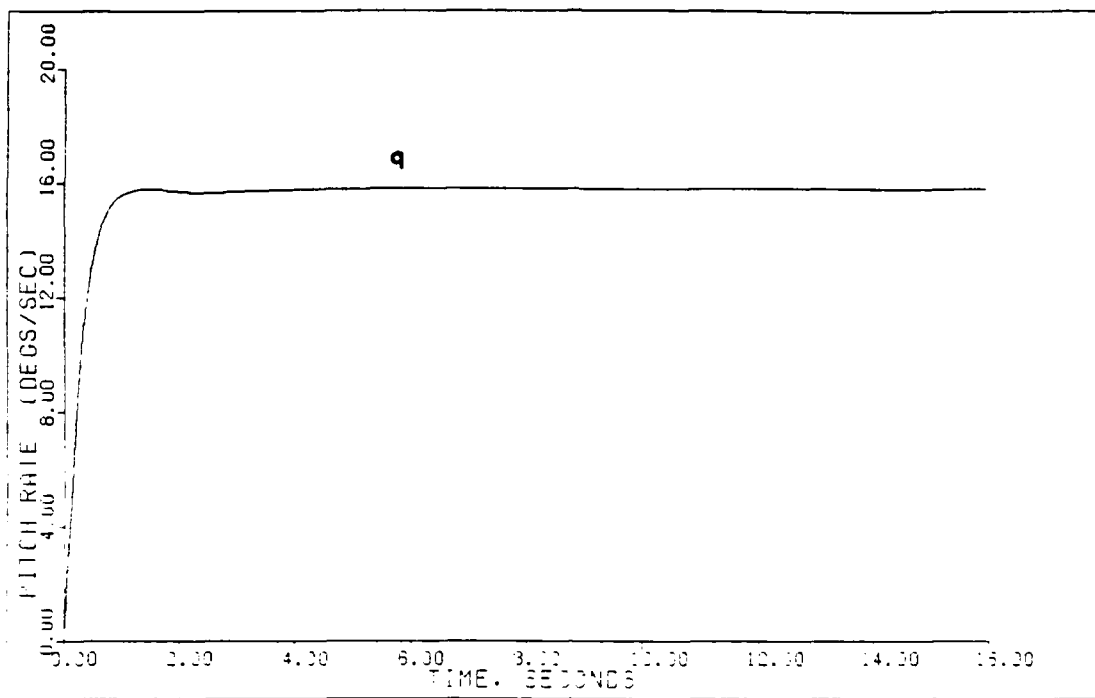
Plant + Actuators

<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
1.00	1.00	1.0	.1148E+02	-.1865E+02
		0.05	-.2562E+01	-.9947E+01

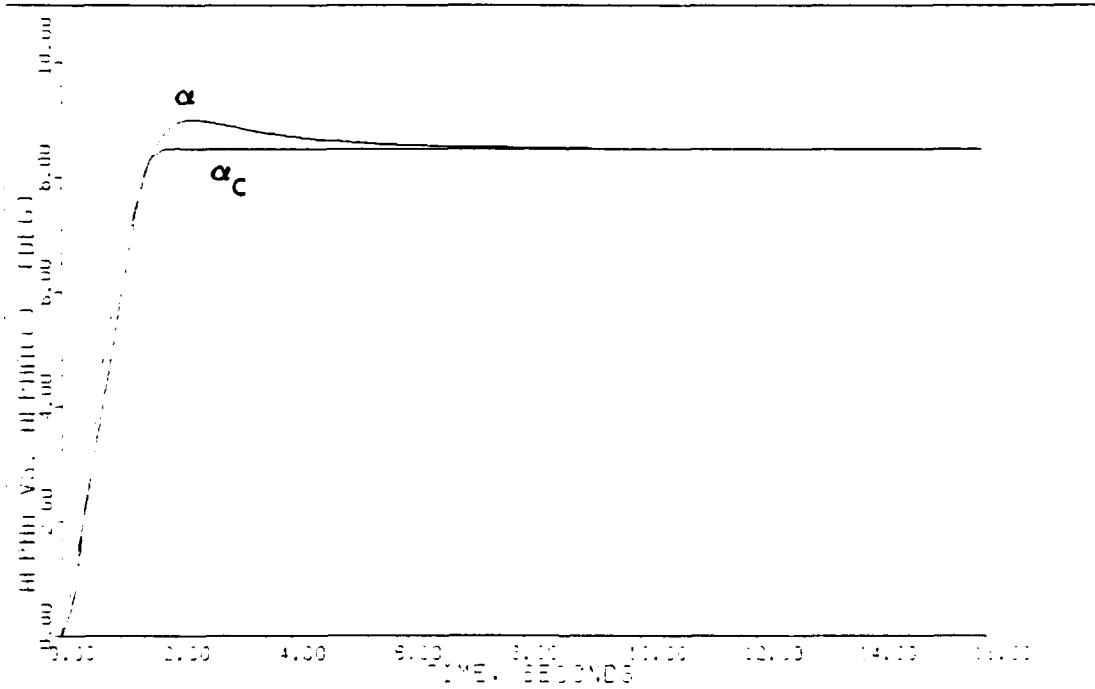
Notes:

- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\text{alpha})\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.

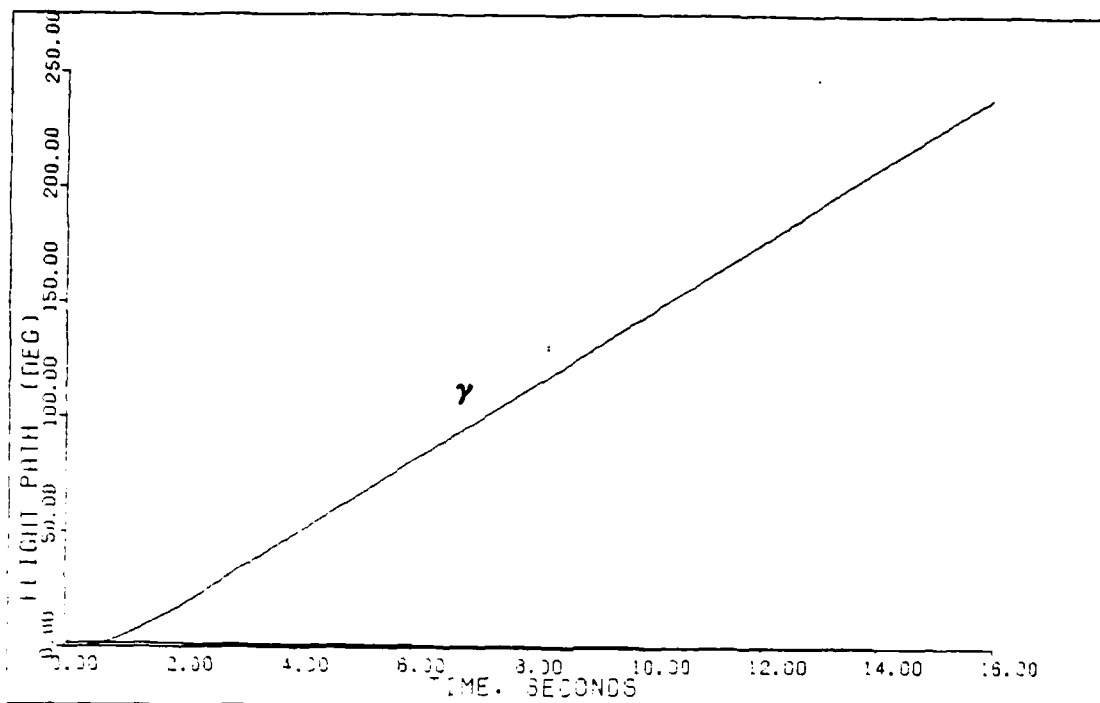




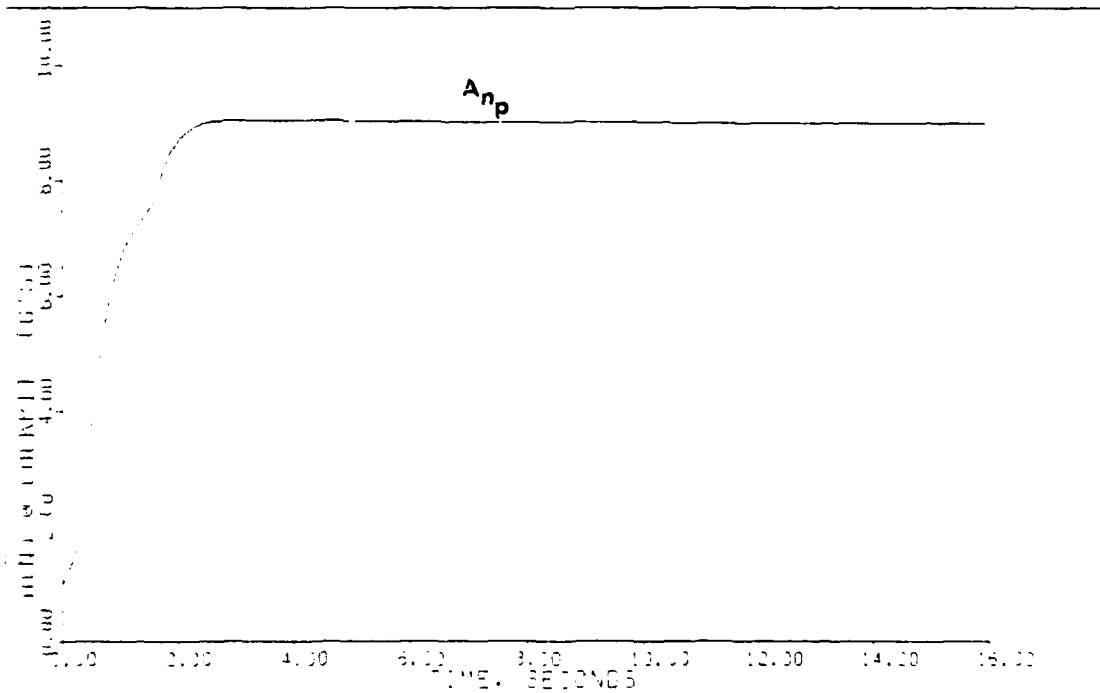
90 PULL-UP: BASIC PLANT 10.9M F1200



90 PULL-UP: BASIC PLANT 10.9M F1200



99 PULL-UP: BASIC PLANT (0.9M/FL200)



99 PULL-UP: BASIC PLANT (0.9M/FL200)

TABLE D.12

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Constant g Pull-up (2.0 g's)

Flight Condition: 1.4 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 20, 0.4435, 20, 20$
 $v_2 = \text{Alpha: } 1.5, 0.008727, 20, 20$

Plant + Actuators

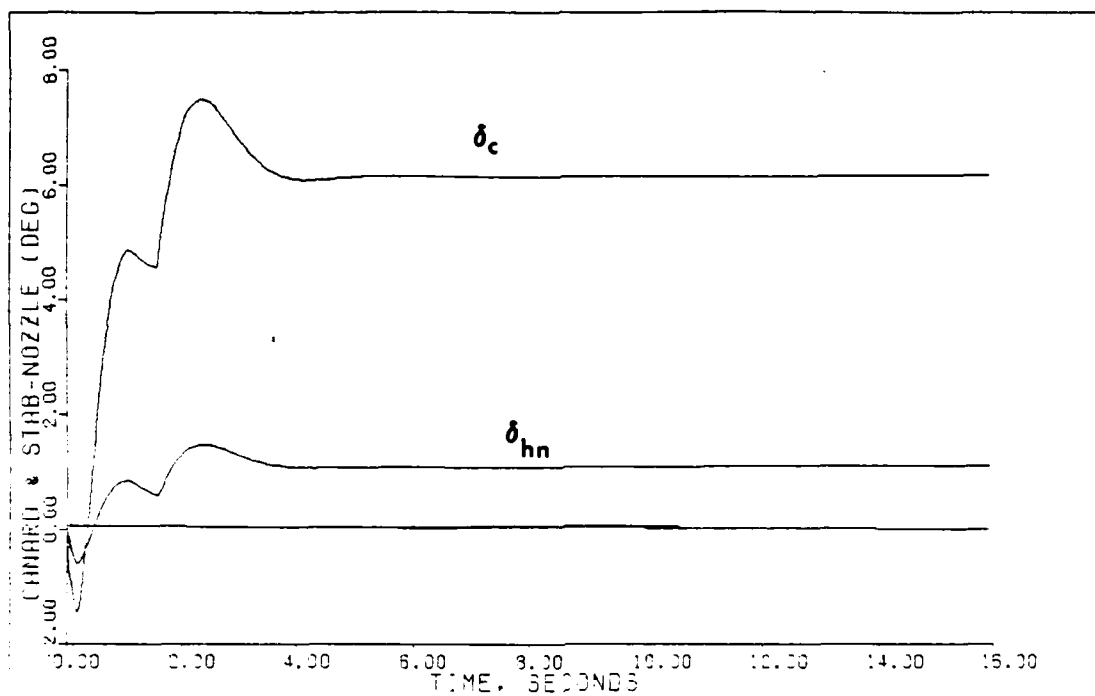
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
2.00	0.50	1.0	.5436E+01	-.7999E+02
		0.25	-.5359E+00	-.2444E+02

Plant + Actuators + Delay

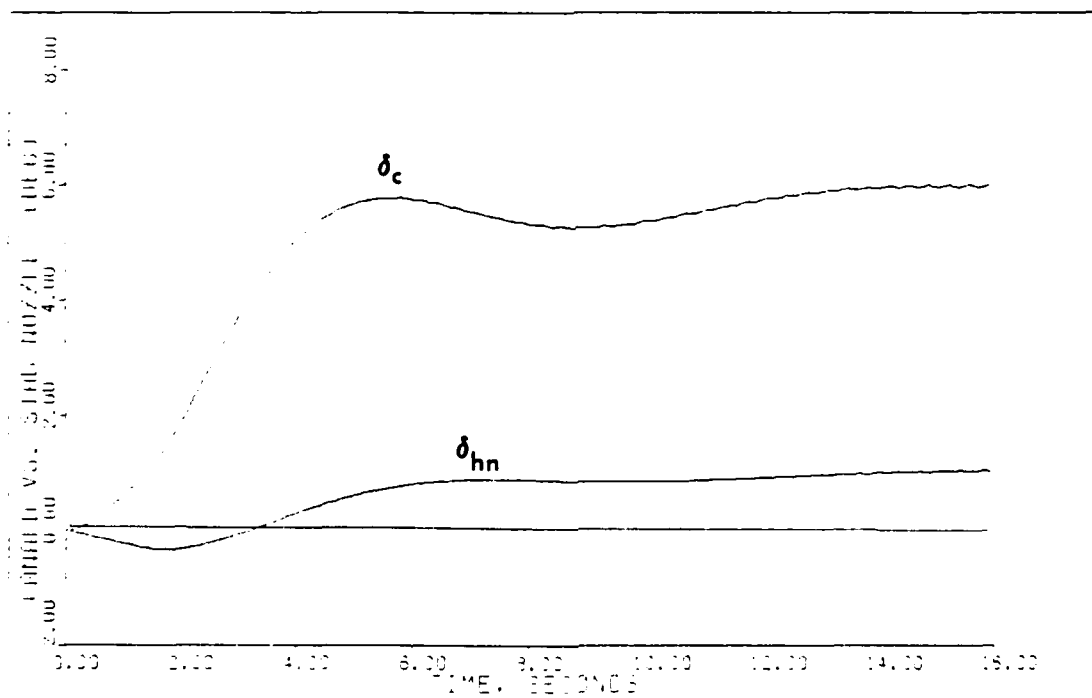
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
2.00	0.200	1.0	.2174E+01	-.5119E+01
		0.04	-.2144E+00	-.1564E+01

Notes:

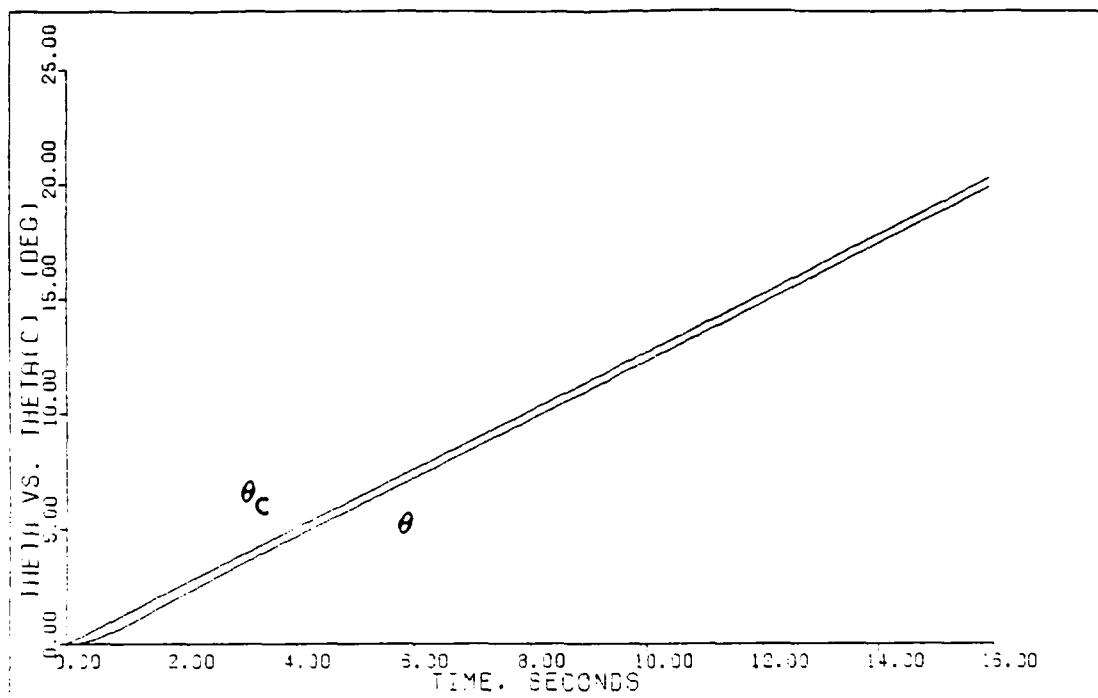
- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\text{alpha})\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



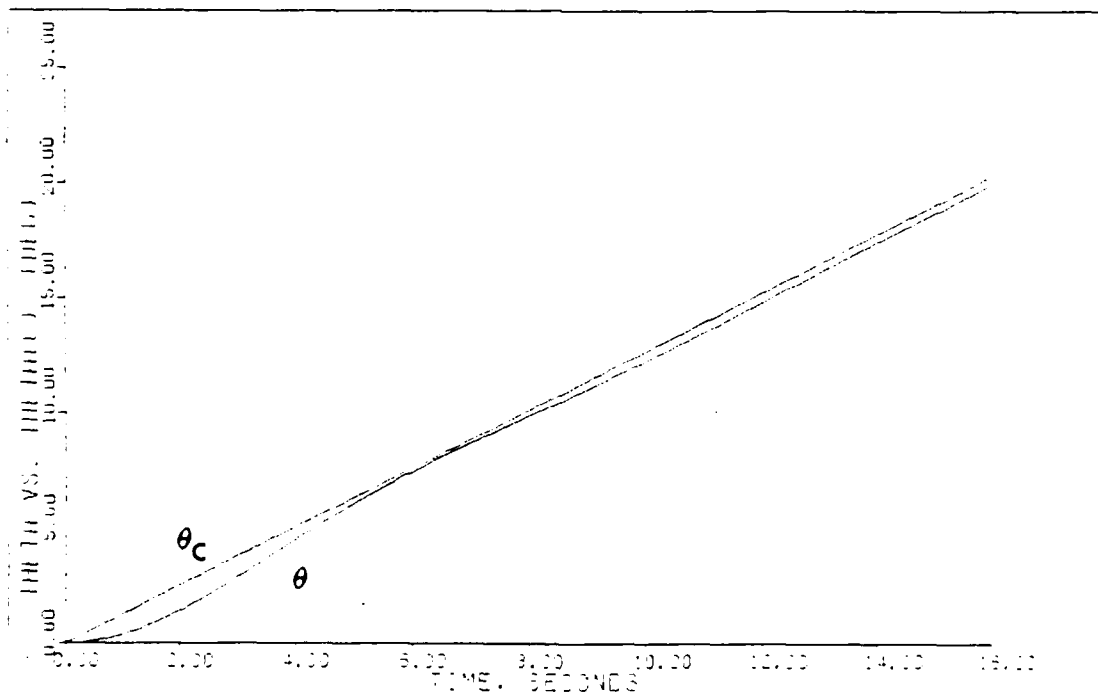
20 PULL-UP: PLANT+ACTUATORS (1.4M/FL200)



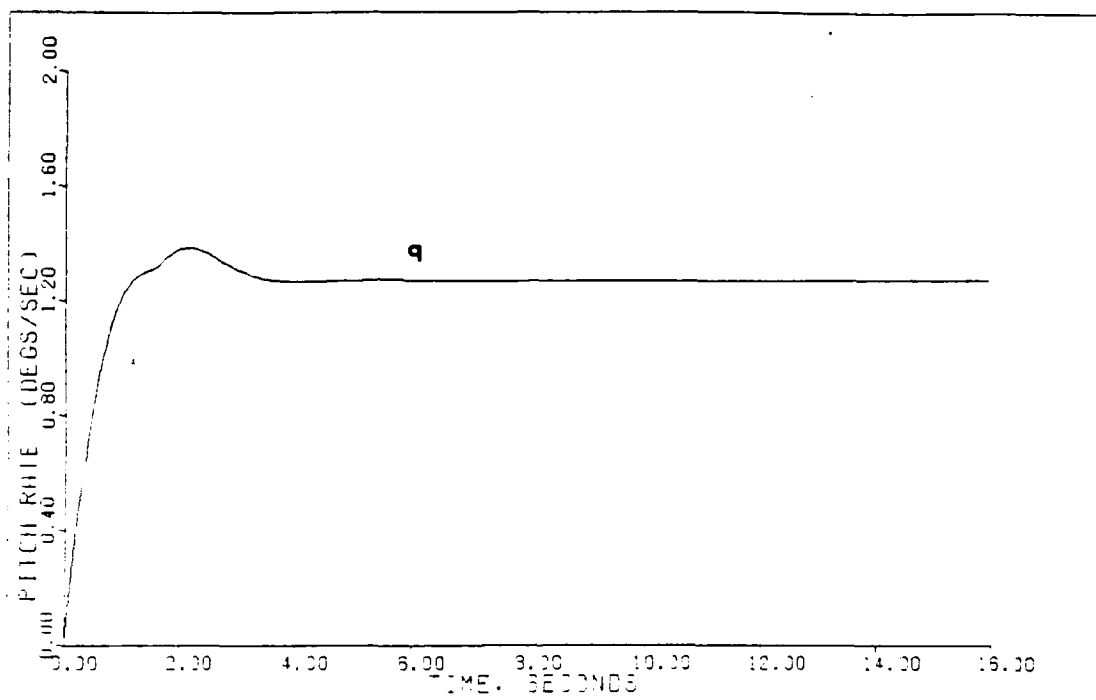
10 PULL-UP: PLANT+ACTUATORS+DELAY (1.4M FL200)



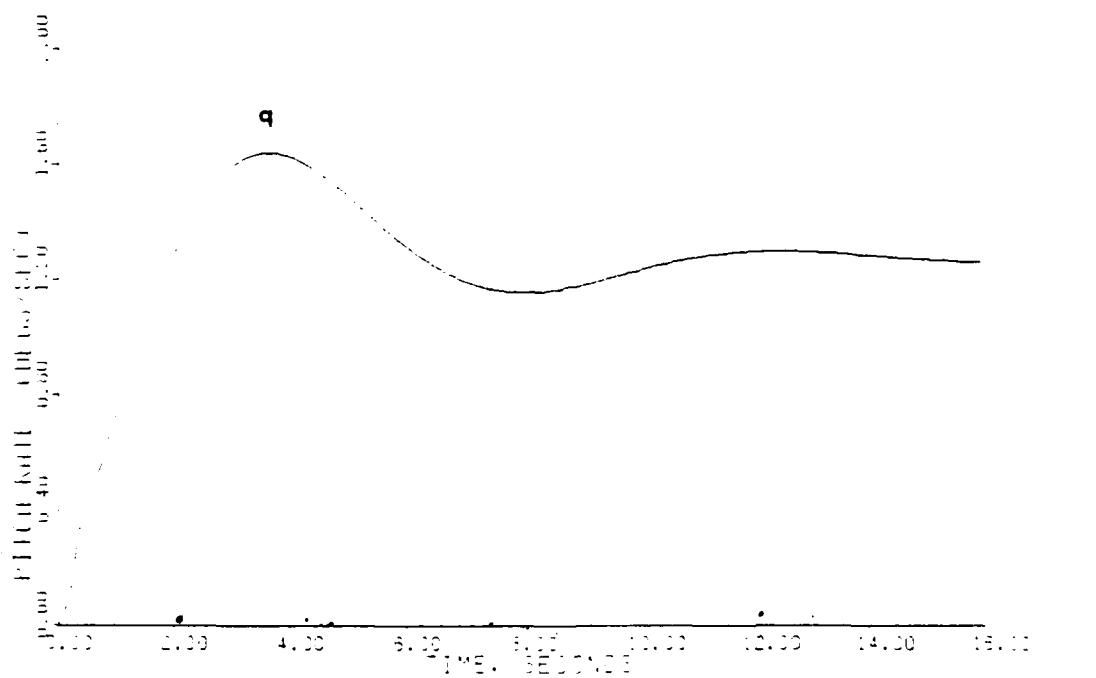
2G PULL-UP: PLANT+ACTUATORS (1.4M/FL200)



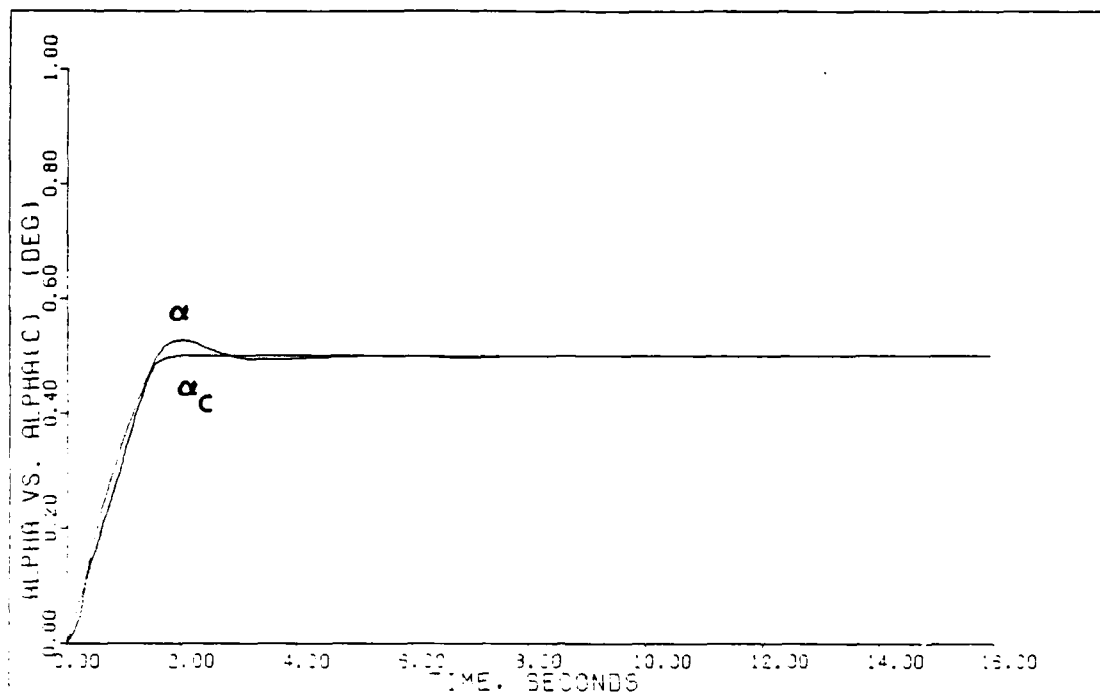
2G PULL-UP: PLANT+ACTUATORS+DELAY (1.4M/FL200)



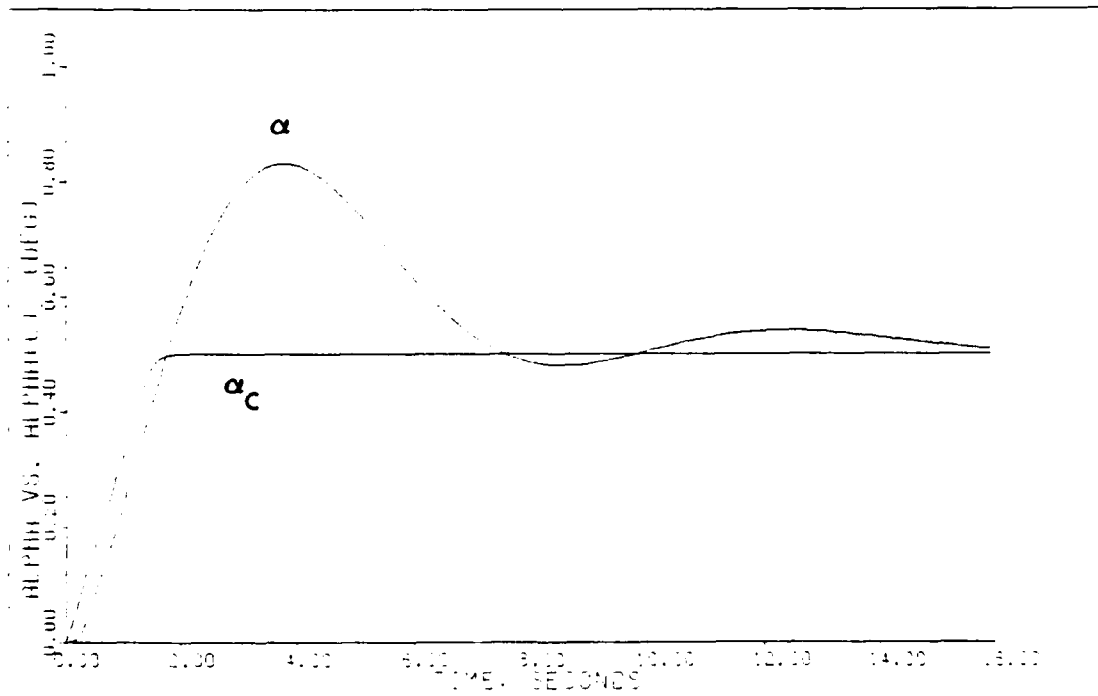
10 PULL-UP: PLANT-ACTUATORS 1.4M/FL200



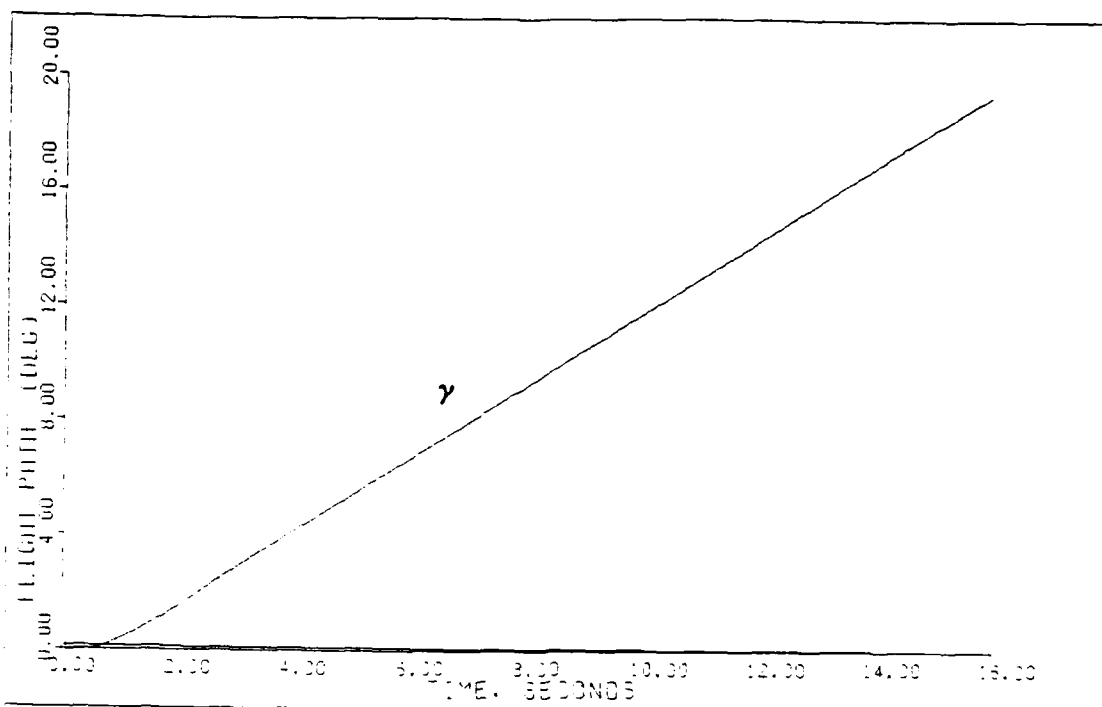
10 PULL-UP: PLANT-ACTUATORS-DELAY 1.4M/FL200



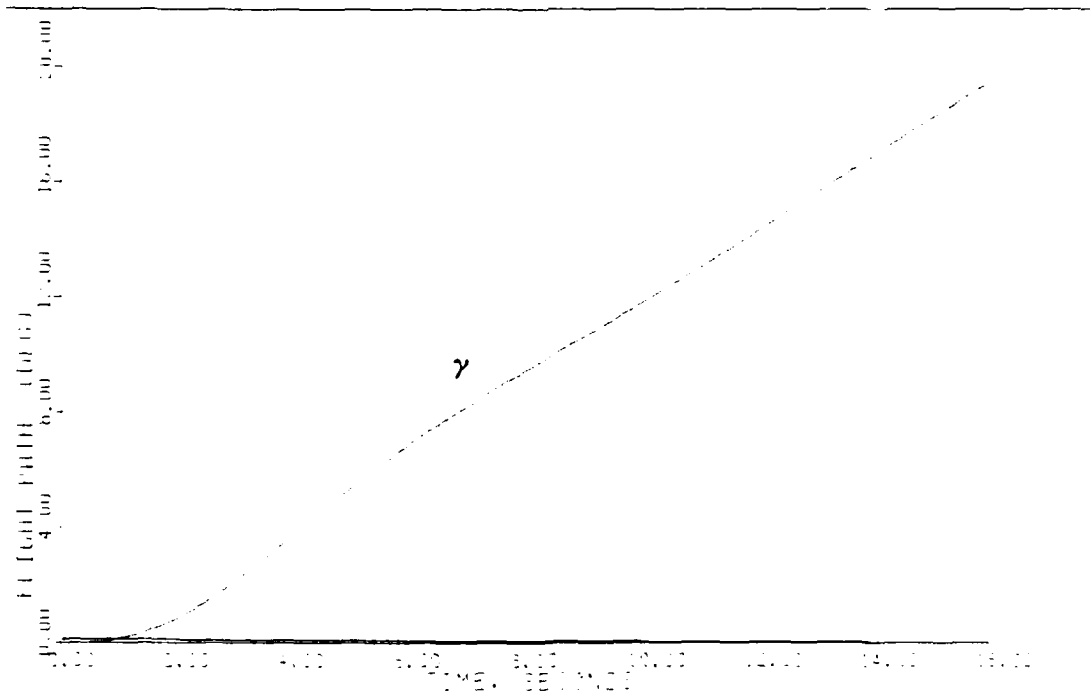
29 PULL-UP: PLANT-ACTUATOR3 (1.4M/FL200)



29 PULL-UP: PLANT-ACTUATOR3+DELAY (1.4M/FL200)



20 PULL-UP PLANT-ROTATORS (11.4M/FLOOD)



20 PULL-UP PLANT-ROTATORS-DELAY (11.4M/FLOOD)

AD-A164 017

MULTIVARIABLE CONTROL LAW DESIGN FOR ENHANCED AIR
COMBAT MANEUVERING F-15. (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. K A SHEEHAN
DEC 85 AFIT/GE/EE/85D-38 F/G 1/2

5/5

UNCLASSIFIED

NL

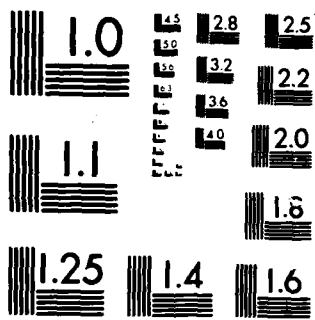


END

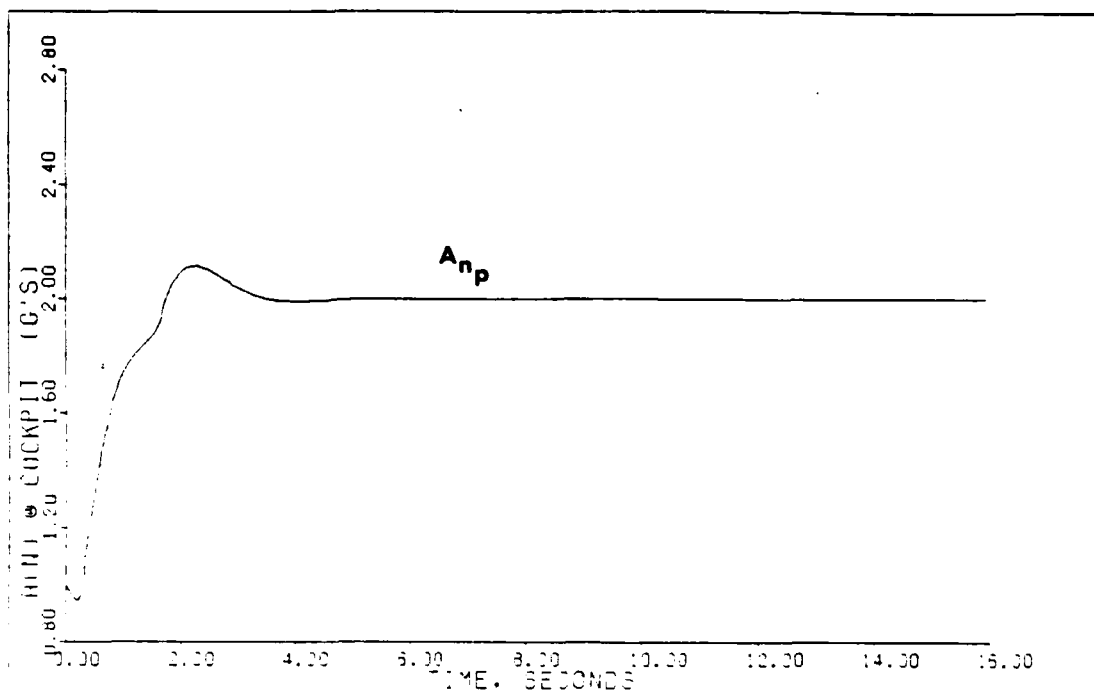
FILMED

1/74

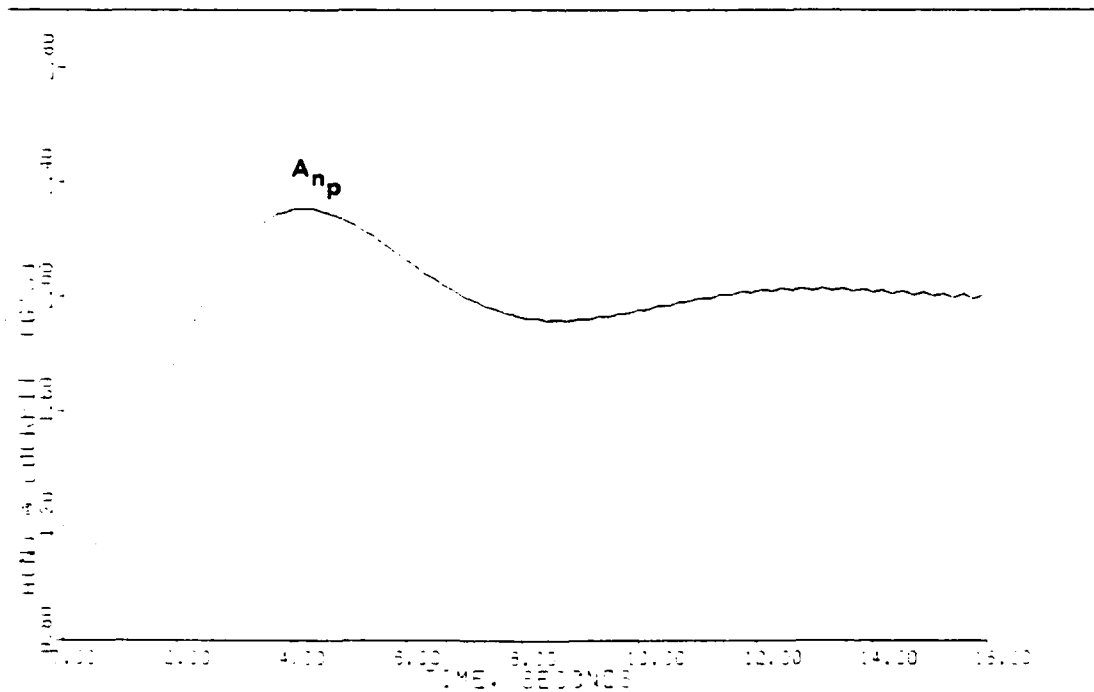
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



29 PULL-UP: PLANT-ACTUATORS (1.4M/FLOOD)



30 PULL-UP: PLANT-ACTUATORS-DELAY (1.4M/FLOOD)

TABLE D.13

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Constant g Pull-up (9.0 g's)

Flight Condition: 1.4 Mach at FL 200

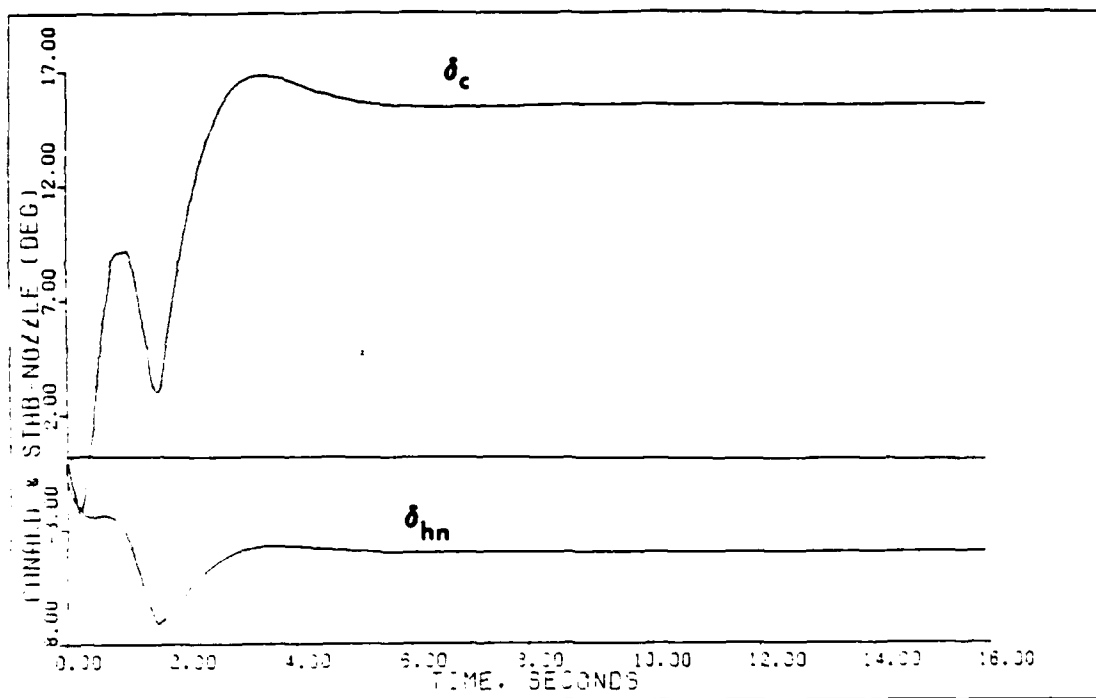
Command Vector \underline{v} : $v_1 = \text{Theta: } 20, 3.54796, 20, 20$
 $v_2 = \text{Alpha: } 1.5, 0.1047198, 20, 20$

Plant + Actuators

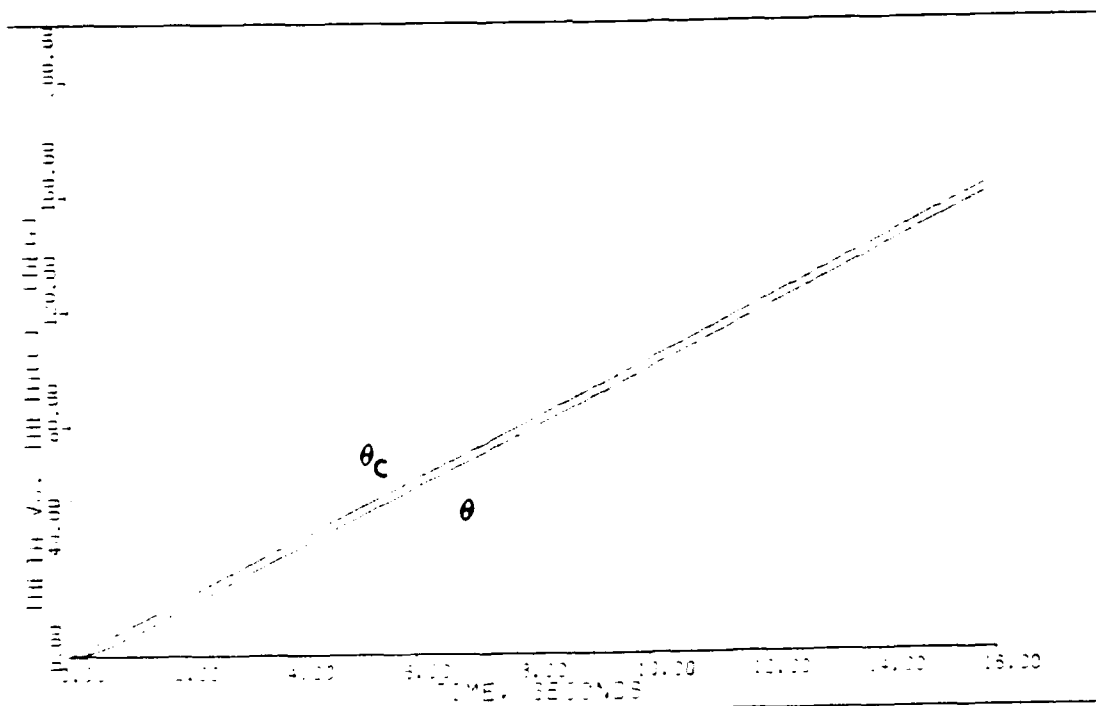
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
2.00	0.50	2.9	.1576E+02	-.1920E+02
		0.06	-.1554E+01	-.5865E+01

Notes:

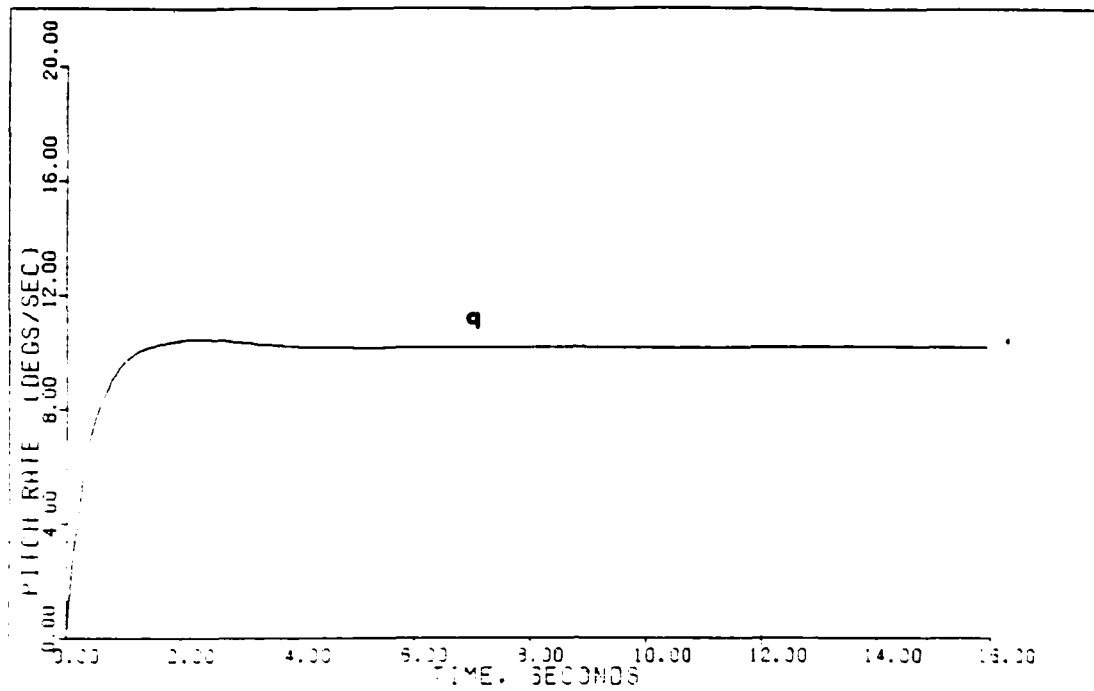
- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\text{alpha})\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



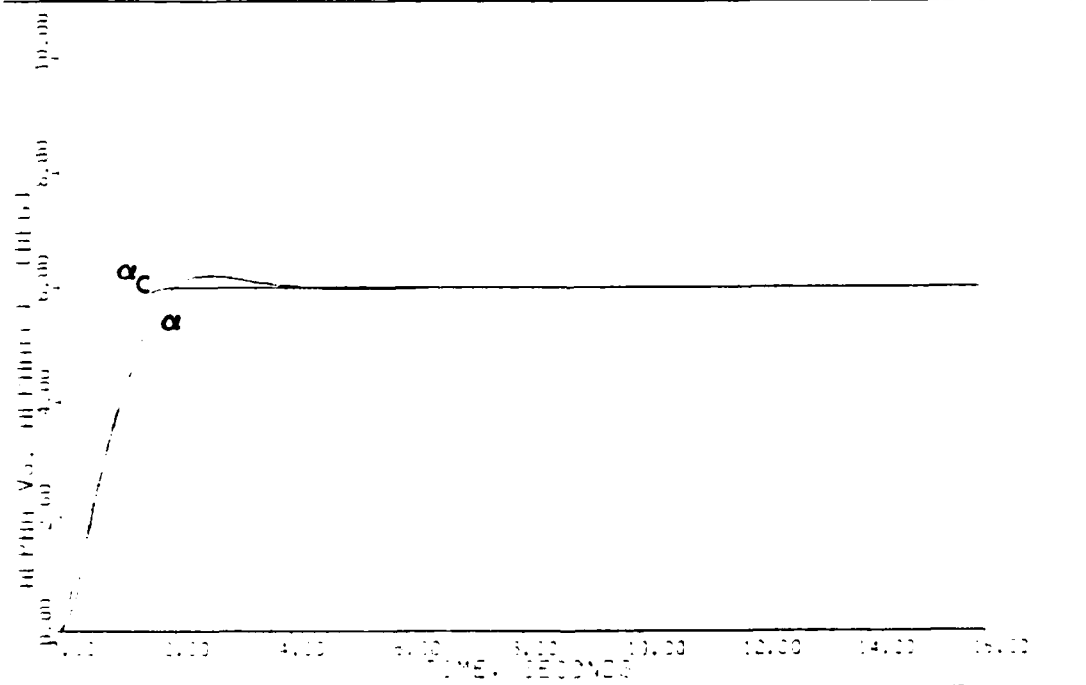
99 PULL-UP: BASIC PLANT (1.4M/FL200)



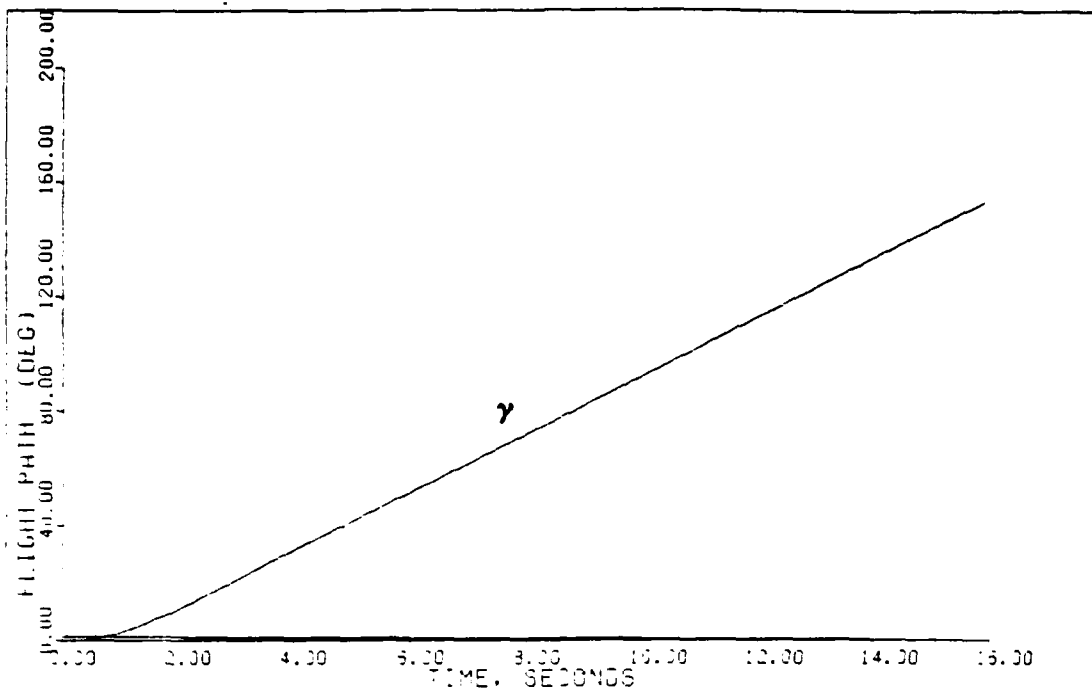
99 PULL-UP: BASIC PLANT (1.4M/FL200)



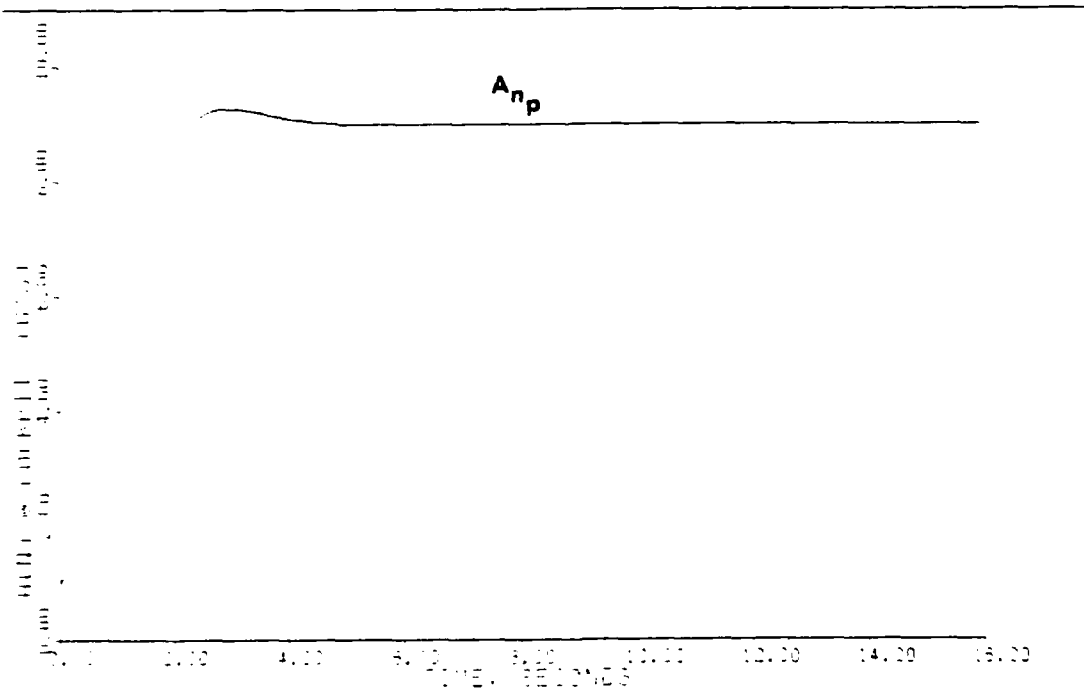
90 FOLLOW-UP: BASIC PLANT (1.4M, FL200)



90 FOLLOW-UP: BASIC PLANT (1.4M, FL200)



30 PULL-UP: BASIC PLANT (1.4M/FL200)



30 PULL-UP: BASIC PLANT (1.4M/FL200)

Bibliography

1. Acker, B. Multivariable Output Feedback Control Law Design for the F-15/STOL in Landing Configuration. MS thesis, AFIT/GE/EE/85D-1. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1985.
2. Barfield, A. Multivariable Control Law Development for the AFTI/F-16. MS thesis, AFIT/GE/EE/84S-4. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1982.
3. Blakelock, J. Automatic Control of Aircraft and Missiles. New York: John Wiley and Sons, Inc., 1965.
4. "British Harriers Average Six Sorties per Day," Aviation Week and Space Technology, 117-3: 20 (Jul 19, 1982).
5. Browne, J. A Piloted Simulation of Direct-Force Control Modes for Air Combat Use, AFWAL-TM-81-85-FIGC. Flight Dynamics Laboratory, Air Force Wright Aeronautical Laboratories, Wright-Patterson AFB OH, July 1981.
6. Courtheyn, Maj T. Multivariable Control Law Design for the X-29 Aircraft. MS thesis, AFIT/GE/EE/84D-21. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1984.
7. D'Azzo, J. and C. Houpis. Linear Control Systems Analysis and Design (Second Edition). New York: McGraw-Hill Book Company, 1981.
8. Eslinger, 2Lt R. Multivariable Control Law Design for the AFTI/F-16 with a Failed Control Surface. MS thesis, AFIT/GE/EE/84D-28. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1984.
9. Etkin, B. Dynamics of Atmospheric Flight. New York: John Wiley and Sons, Inc., 1972.
10. Farrell, J. Integrated Aircraft Navigation. New York: Academic Press, 1976.

11. Heller, W. Models for Aided Inertial Navigation System Sensor Errors. Massachusetts: Analytical Science Corporation, 1975.
12. Houppis, C. and G. Lamont. Digital Control Systems Theory, Hardware, Software. New York: McGraw-Hill Book Company, 1985.
13. Masi, A. and D. Russ. Multi User's Manual. AFWAL-TM-83-182-FIGL. Air Force Flight Dynamics Laboratory, Wright-Patterson AFB OH, August 1984.
14. Maybeck, P. Stochastic Models, Estimation, and Control, Volume 1. New York: Academic Press, 1979.
15. McDonnell Aircraft Co. Subcontractor Engineering Memorandum, SMTP-HSR-001, 10 Jan 1985.
16. "McDonnell Douglas to Develop F-15/STOL," Aviation Week and Space Technology, 122-15: 21 (Oct 8, 1984).
17. Moseley, A. Design of Advanced Digital Flight Control Systems via Command Generator Tracker (CGT) Synthesis Methods. Volume 1. MS thesis, AFIT/GE/EE/82D-51. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1982.
18. Porter, B. and A. Bradshaw. "Design of Linear Multi-variable Discrete-Time Tracking Systems Incorporating Fast-Sampling Error-Actuated Controllers," International Journal of System Science, 2 (7): 817-826 (1980).
19. ----- "Singular Perturbation Methods in the Design of Tracking Systems Incorporating High-Gain Error Actuated Controllers," International Journal of System Science, 12 (10): 1169-1220 (1981).
20. Pugh, A. "Transmission and System Zeros," International Journal of Control, 26 (2): 315-324 (1977).
21. Roskam, J. Airplane Flight Dynamics and Automatic Flight Controls. Lawrence KS: Roskam Aviation and Engineering Corporation, 1976.
22. "V/STOL Propulsion: Past, Present, and Future," Air Force Magazine, 65: 76-79 (Jan 1982).

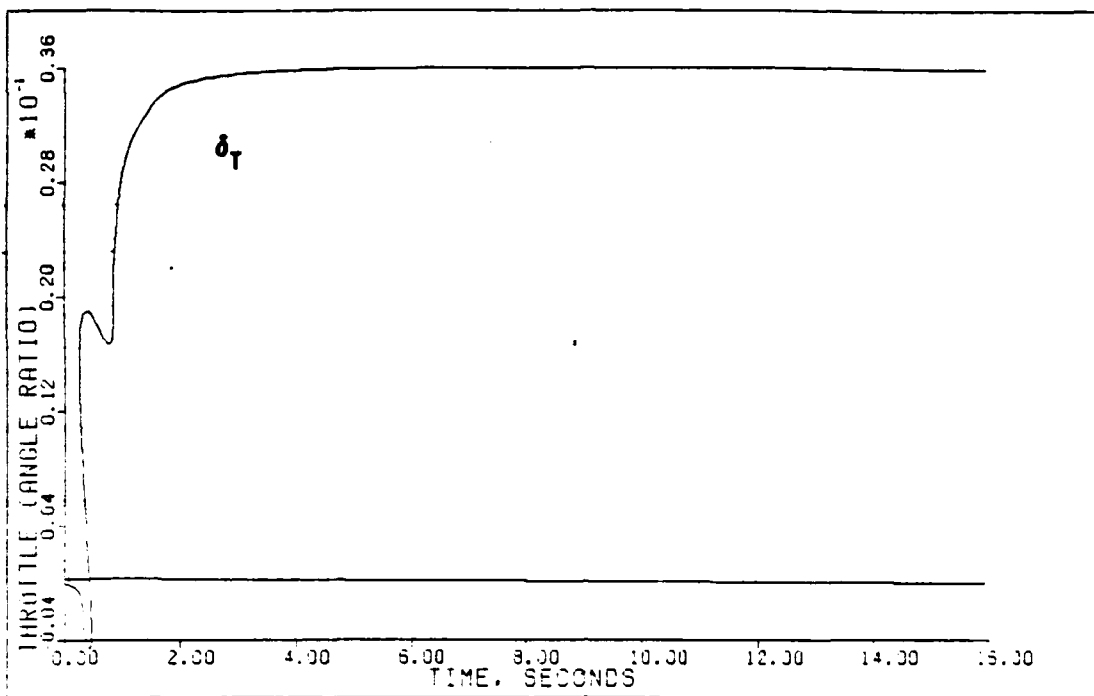
VITA

Captain Kevin Sheehan is a native of Chicago, Illinois. He graduated from the University of Notre Dame, Notre Dame, Indiana in 1975 with a Bachelor of Science in Electrical Engineering. He was then commissioned through the Air Force ROTC program and entered Undergraduate Pilot Training in June of 1976 at Williams AFB, Phoenix, Arizona.

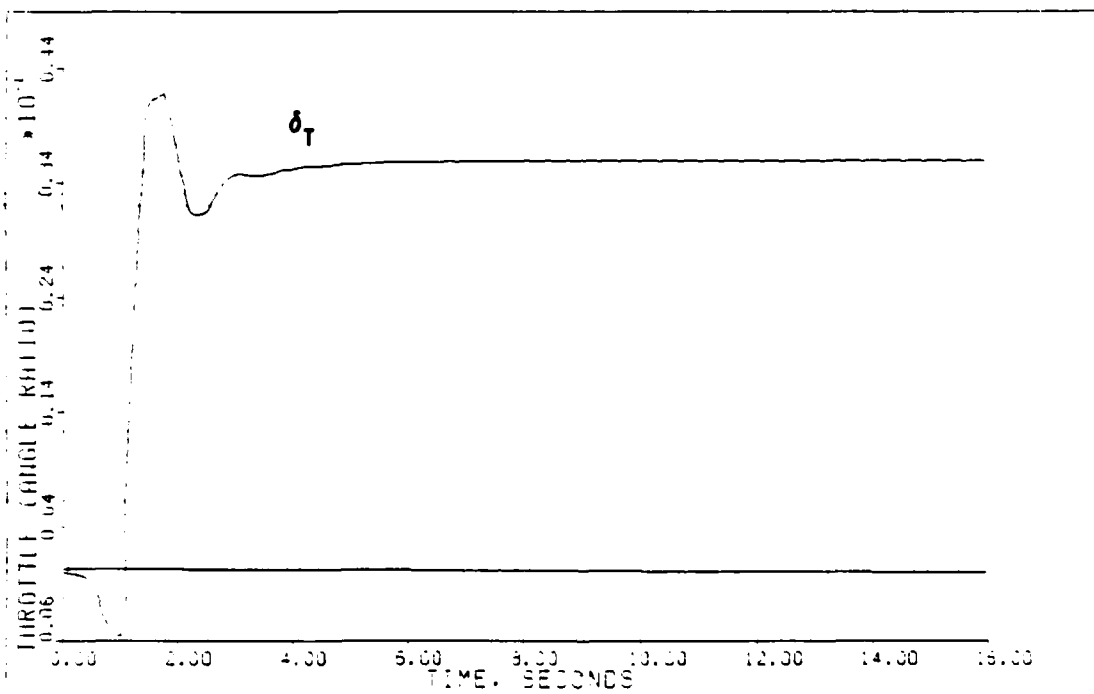
He remained at Williams AFB as a T-38 Instructor Pilot until 1981 when he received an assignment to fly the F-15 at Bitburg AB, West Germany. Captain Sheehan worked as an Assistant Flight Commander and aircraft Flight Lead logging over 600 hours of air-to-air experience in the F-15 Eagle. While stationed at Bitburg, he attended night school and received a Master of Science in Management from Troy State University, Troy, Alabama.

In May of 1984, he was selected to attend the Air Force Institute of Technology, Wright-Patterson AFB, Ohio. He received a Master of Science in Electrical Engineering, specializing in the area of Guidance and Control, in December 1985. Currently, he is assigned to the Tactical Air Warfare Center, Eglin AFB, Florida.

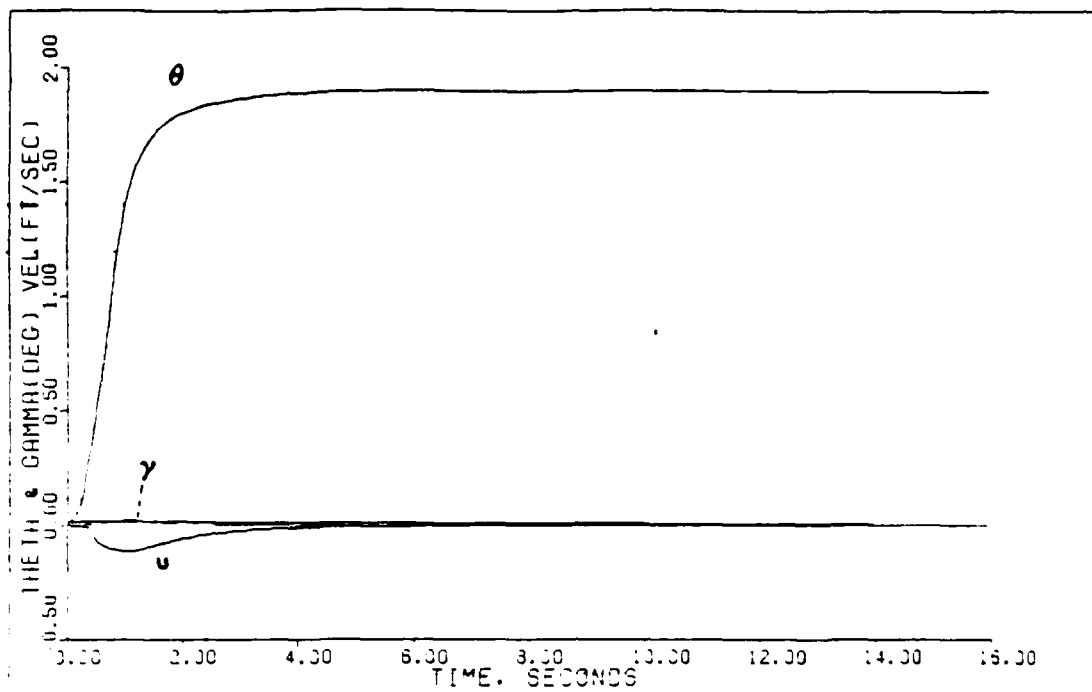
Permanent address: 1570 Stonehill Court
Wheaton, Illinois 60187



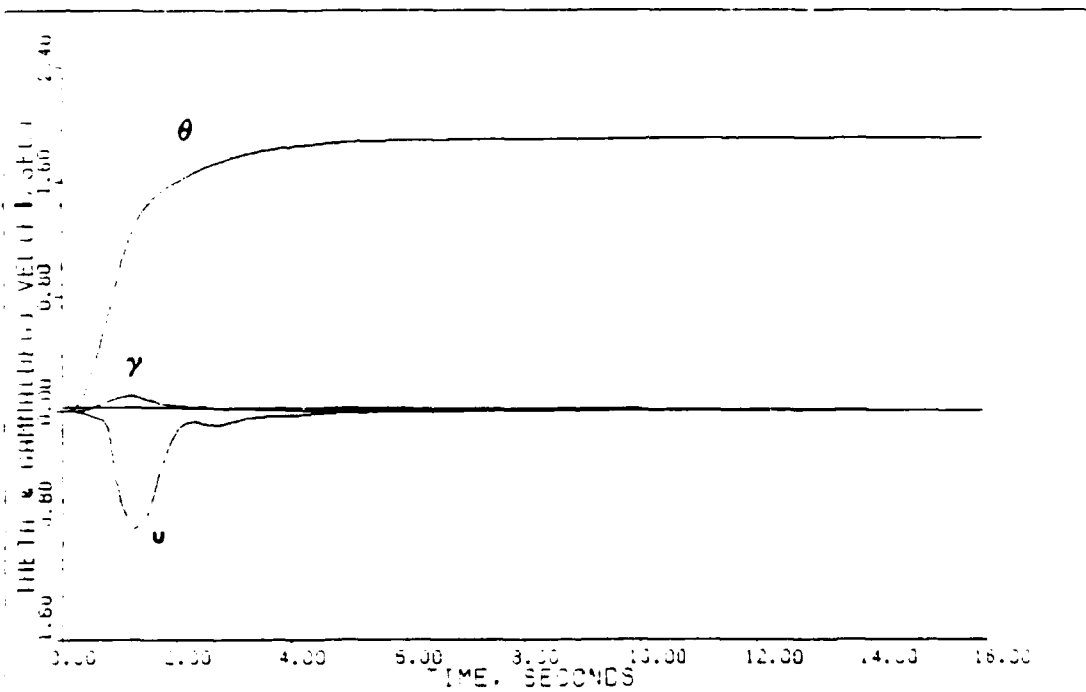
PITCH POINTING: PLANT-ACTUATORS (2.0M/FL400)



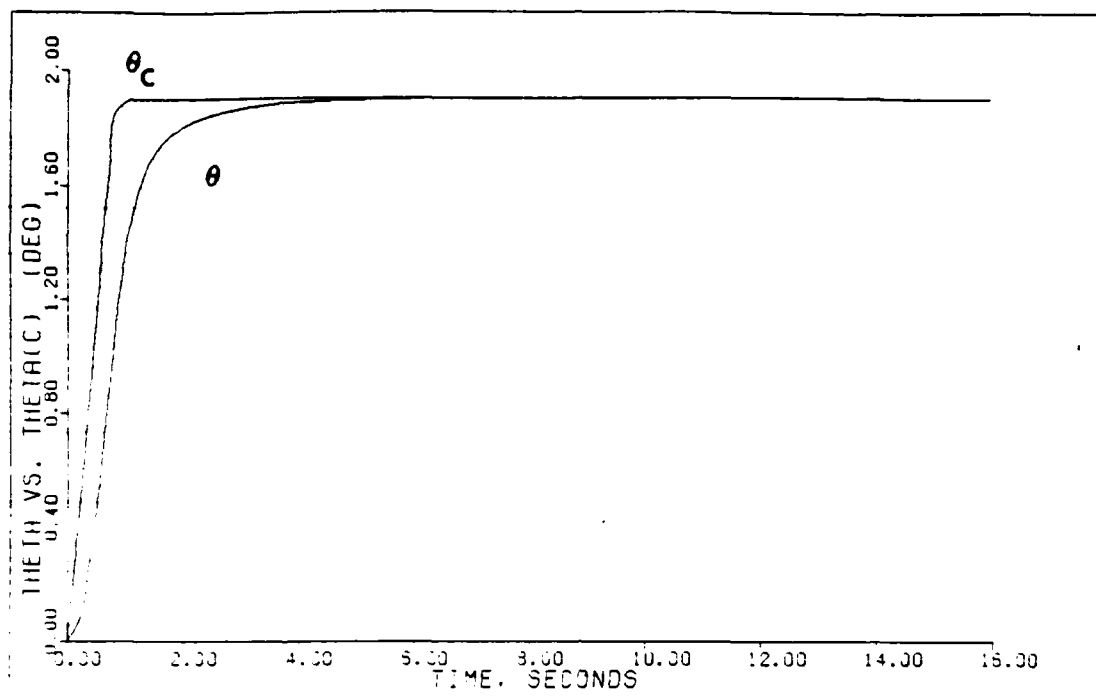
PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (2.0M/FL400)



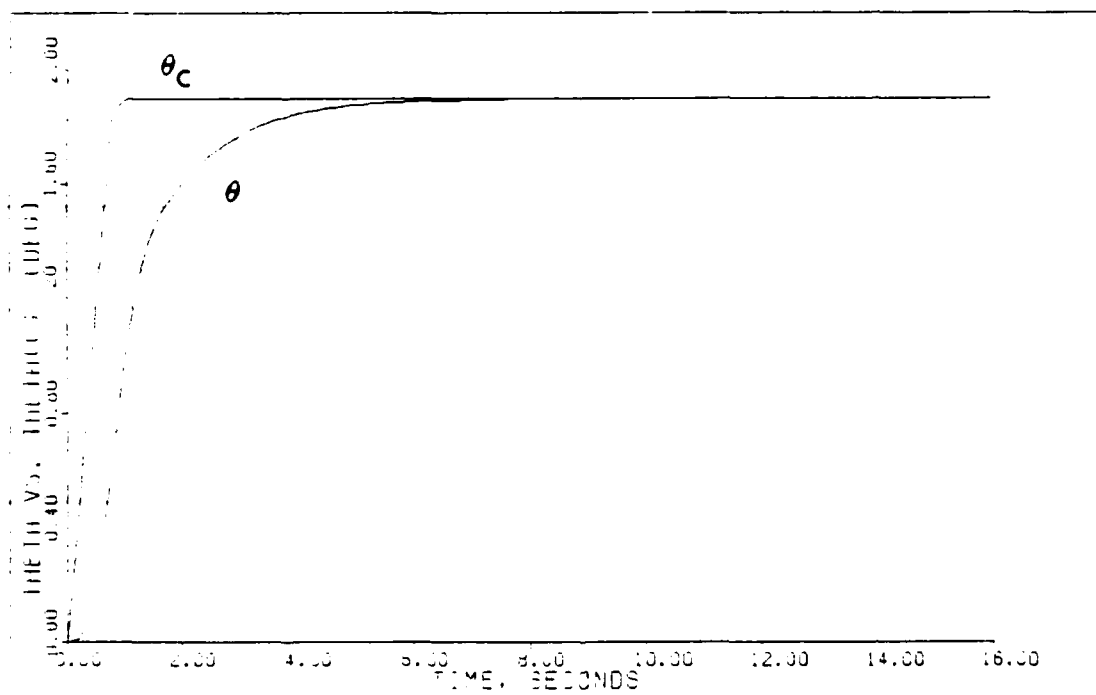
PITCH POINTING: PLANT-ACTUATORS (2.0M/FL400)



PITCH POINTING: PLANT-ACTUATORS-DELAY-SENSORS (2.0M/FL400)



PITCH POINTING: PLANT+ACTUATORS (2.0M/FL400)



PITCH POINTING: PLANT+ACTUATORS+DELAY+SENSORS (2.0M/FL400)

TABLE D.10

DESIGN PARAMETERS AND CONTROLLER MATRICES

Maneuver: Constant g Pull-up (2.0 g's)

Flight Condition: 0.9 Mach at FL 200

Command Vector \underline{v} : $v_1 = \text{Theta: } 20, 0.6899, 20, 20$
 $v_2 = \text{Alpha: } 1.5, 0.008727, 20, 20$

Plant + Actuators

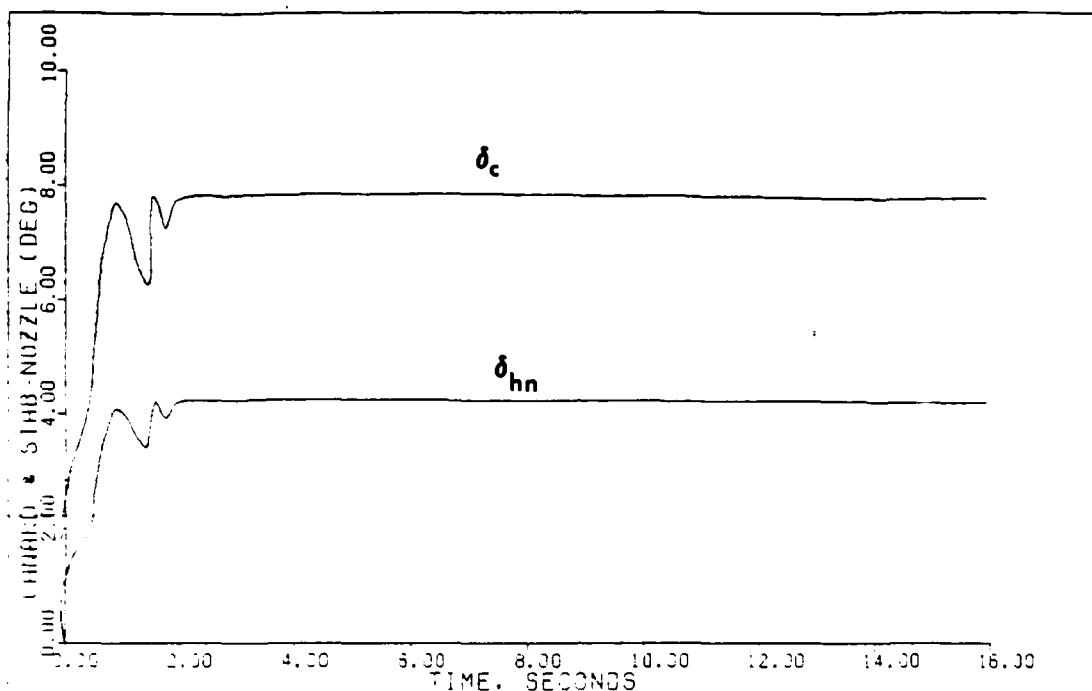
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
1.00	1.00	1.0	.1148E+02	-.3730E+03
		1.0	-.2562E+01	-.1989E+03

Plant + Actuators + Delay + Sensors

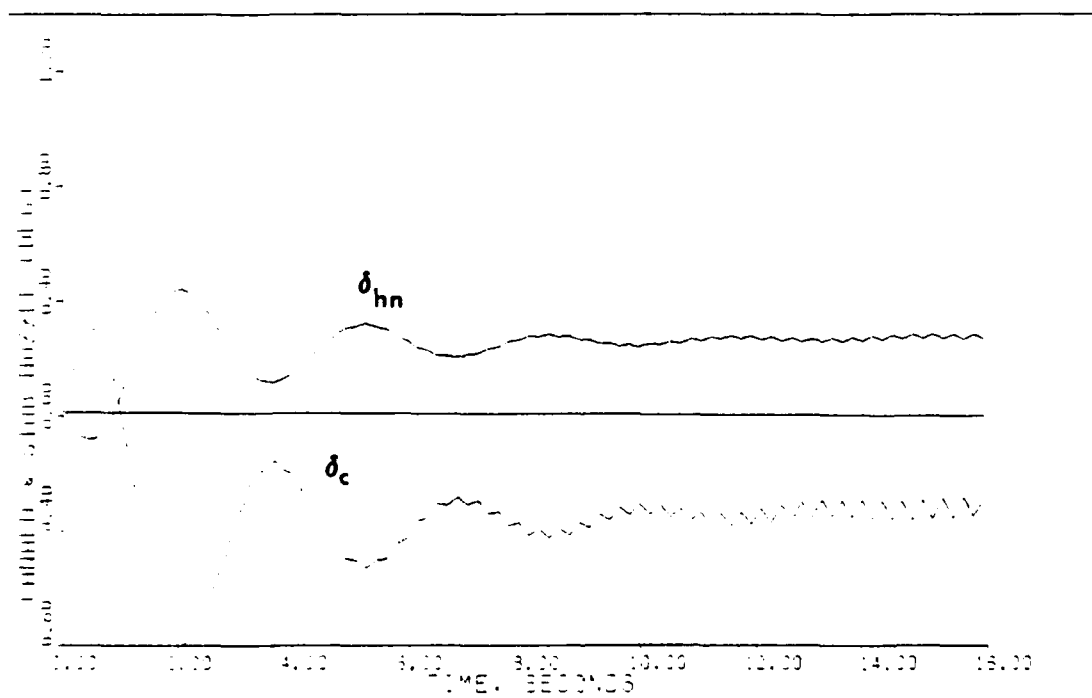
<u>Alpha</u>	<u>Epsilon</u>	<u>Sigma</u>	<u>\underline{K}_0</u>	
0.033	0.180	1.0	.2067E+01	-.6714E+00
		0.01	-.4611E+00	-.3581E+00

Notes:

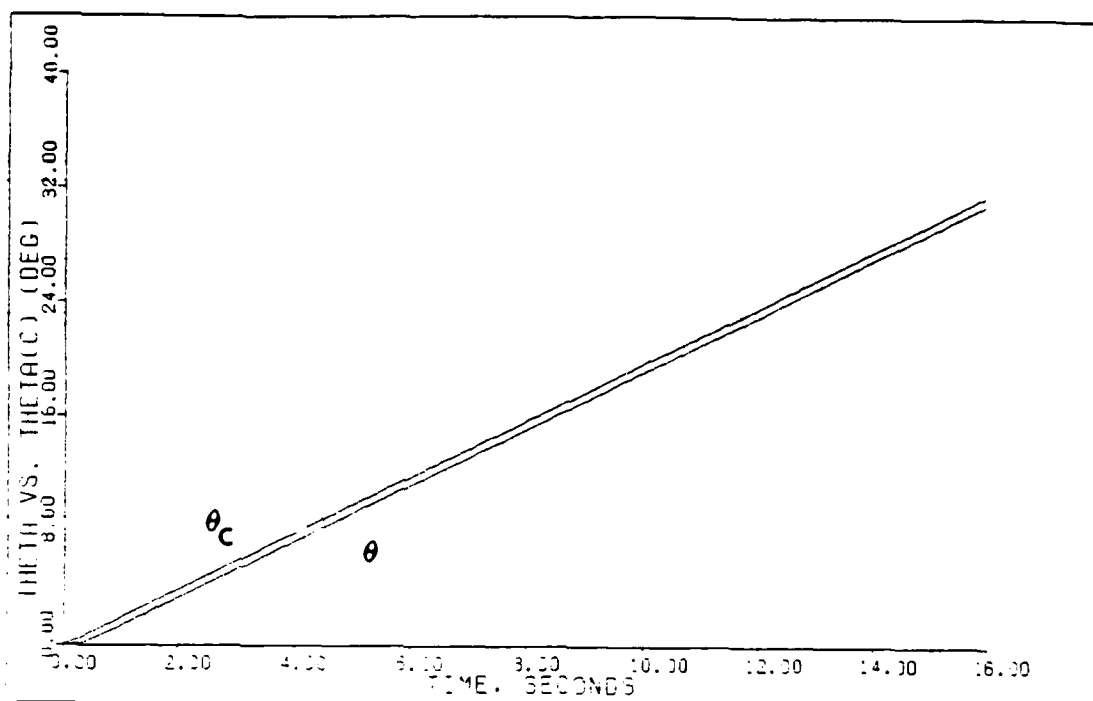
- Each \underline{v} input is composed of four parts:
 - Time (secs) that the input reaches steady-state.
 - Steady-state value (radians).
 - Time (secs) input leaves steady-state.
 - Time (secs) input reaches zero.
- Sigma = the elements (in order) of the diagonal matrix.
- The integral controller matrix $\underline{K}_1 = (\text{alpha})\underline{K}_0$.
- Irregular design: $\underline{M} = \{0.3, 0, 0\}^T$.



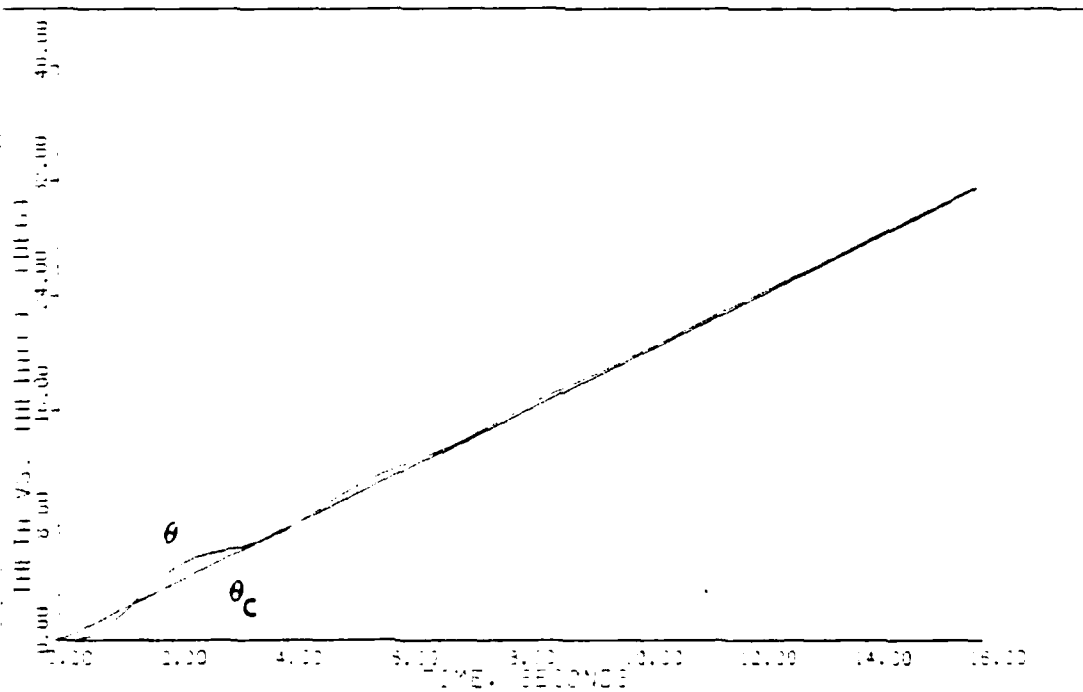
CG PULL-UP: PLANT-ACTUATORS (0.9M/FL200)



CG PULL-UP: PLANT-ACTUATORS-DELAY-SENSORS (0.9M/FL200)



10 PULL-UP: PLANT-ACTUATORS (0.9MFL200)



10 PULL-UP: PLANT-ACTUATORS-DELAY-SENSORS (0.9MFL200)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD-A164017

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			Approved for public release; distribution unlimited	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GE/EE/85D-38			5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION School of Engineering		6b. OFFICE SYMBOL (If applicable) AFIT/ENG	7a. NAME OF MONITORING ORGANIZATION	
6c. ADDRESS (City, State and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB OH 45433			7b. ADDRESS (City, State and ZIP Code)	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Flight Dynamics Lab.		8b. OFFICE SYMBOL (If applicable) AFWAL/FIGX	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c. ADDRESS (City, State and ZIP Code) Wright-Patterson AFB, OH 45433			10. SOURCE OF FUNDING NOS.	
11. TITLE (Include Security Classification) See Box 19			PROGRAM ELEMENT NO.	TASK NO.
			PROJECT NO.	WORK UNIT NO.
12. PERSONAL AUTHOR(S) Kevin A. Sheehan, B.S.E.E., Captain, USAF				
13a. TYPE OF REPORT MS Thesis		13b. TIME COVERED FROM _____ TO _____		14. DATE OF REPORT (Yr., Mo., Day) 1985 December
				15. PAGE COUNT 399
16. SUPPLEMENTARY NOTATION Approved for public release; IAW AFR 190-17				
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB. GR.	Multivariable Control, Flight Control System, Digital Control System, Thrust Vectoring, STOL Technology	
01	03			
19. ABSTRACT (Continue on reverse if necessary and identify by block number)				
Title: MULTIVARIABLE CONTROL LAW DESIGN FOR ENHANCED AIR COMBAT MANEUVERING: F-15/STOL DERIVATIVE FIGHTER				
Thesis Chairman: John J. D'Azzo				
Approved for public release: IAW AFR 190-17. Lyle E. WOIWAGER 16 JAN 86 Dean for Research and Professional Development Air Force Institute of Technology (AFIT) Wright-Patterson AFB OH 45433				
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS <input type="checkbox"/>			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL			22b. TELEPHONE NUMBER (Include Area Code)	22c. OFFICE SYMBOL

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

Digital flight control laws which demonstrate improved air combat maneuverability are developed for the F-15/STOL (Short Take-Off and Landing) derivative fighter. Proportional plus integral controllers are designed for the longitudinal mode using a multivariable control law theory developed by Professor Brian Porter of the University of Salford, England. Control laws are formulated by use of a computer-aided, multivariable design program entitled MULTI. In addition, MULTI performs a digital closed-loop simulation for controller performance analysis.

The aircraft model is developed from linearized data provided by McDonnell Aircraft Co., the prime contractor for the F-15/STOL. Canard and thrust vectoring technology, in addition to conventional control surfaces, are included in the model. Decoupling of the longitudinal output variables is achieved and demonstrated by four maneuvers (pitch-pointing, vertical translation, direct climb, and constant g pull-up). Plant parameter variation effects are also examined. Destabilizing effects to include actuator and sensor dynamics, computational time delay, random Gaussian sensor noise, and simulation nonlinearities are included.

Results show stable responses for all simulations. Except for the most demanding simulations (all destabilizing effects considered), controller responses are smooth and well behaved.

Recommendations include proposed future work in thrust vector modeling and suggested improvements to the computer-aided design program, MULTI.

UNCLASSIFIED

END

FILMED

386

DTIC